



UNIVERSIDAD DE LA RIOJA

TRABAJO FIN DE ESTUDIOS

Título

Aplicación web para la gestión de proyectos I+D

Autor/es

YACY LODOS SARANOVA

Director/es

JESÚS MARÍA ARANSAY AZOFRA

Facultad

Facultad de Ciencia y Tecnología

Titulación

Grado en Ingeniería Informática

Departamento

MATEMÁTICAS Y COMPUTACIÓN

Curso académico

2018-19



Aplicación web para la gestión de proyectos I+D, de YACY LODOS SARANOVA (publicada por la Universidad de La Rioja) se difunde bajo una Licencia Creative Commons Reconocimiento-NoComercial-SinObraDerivada 3.0 Unported. Permisos que vayan más allá de lo cubierto por esta licencia pueden solicitarse a los titulares del copyright.



UNIVERSIDAD DE LA RIOJA

Facultad de Ciencia y Tecnología

TRABAJO FIN DE GRADO

Grado en Ingeniería Informática

Aplicación web para la gestión de proyectos de I+D

Realizado por:

Yacy Lodos Saranova.

Tutelado por:

Jesús María Aransay Azofra.

Logroño, 23 de Julio de 2019

Resumen

En este trabajo fin de grado, he desarrollado una aplicación para la gestión de proyectos de I+D. Cumpliendo los requisitos especificados por el cliente en esta aplicación, se pueden gestionar los proyectos desde su inicio hasta su fin siguiendo el proceso de trabajo de Ecoembes. Este proceso implica la gestión de presupuestos (adjudicaciones y facturaciones), gestión de documentos internos, gestión de tareas mediante un tablero Kanban, gestión de empleados y visualización de gráficos con los datos de proyectos y datos de dedicación de los empleados.

Abstract

In this final degree project, I have developed an application for the management of I&D projects. Fulfilling the specified requirements by the client in this application, projects can be managed from the beginning to the end following the Ecoembes work process. This process involves the management of budgets (awards and billing), internal document management, task management through a Kanban board, employee management and graphic visualization with project data and employee dedication data.

Índice

Resumen	3
Abstract	3
Índice	4
1.Introducción	5
1.1. Actores	5
1.2. Análisis de requisitos	5
1.3. Análisis de Mercado	8
1.4. Metodología	9
1.5. EDT	11
1.6. Gantt	11
1.7. Product Backlog/ pila de producto	13
1.8. Entregables	22
1.9. Exclusiones	22
2. Planificación	23
2.1. Sprint 0	23
2.2. Sprint 1	23
2.3. Sprint 2	25
2.4. Sprint 3	27
2.5. Sprint 4	29
2.6. Sprint 5	31
3. Sprint	33
3.1. Sprint 1	33
3.2. Sprint 2	38
3.3. Sprint 3	42
3.4. Sprint 4	46
3.5. Sprint 5	52
4. Revisión de la gestión	55
5. Conclusiones	55
Bibliografía	56
Anexo	57

1.Introducción

Ecoembes [1] es una organización sin ánimo de lucro que se dedica a la recuperación de envases en España. Crean en la eficiencia medioambiental y económica y trabajan para conseguir que el reciclaje de envases de España crezca cada año.

Un pilar básico de Ecoembes es TheCircularLab (TCL), primer centro de innovación sobre economía circular creado en Europa que persigue impulsar las mejores líneas de innovación en el ámbito de los envases y su posterior reciclado. Estudia, concibe, prueba y aplica en un entorno real, las mejores prácticas en todas las fases del ciclo de vida de los envases, desde su concepción hasta su reintroducción en el ciclo de consumo a través de nuevos productos.

La metodología con la que trabajan en TCL es una metodología ágil, la cual intentan seguir en los procesos que conlleva un proyecto. Los trabajadores de Ecoembes de TCL trabajan en muchos proyectos conjuntos con sus compañeros de Ecoembes de otras localidades, para ello es de gran ayuda las herramientas de trabajo ágil. En la actualidad, cuentan con una aplicación web para la gestión de proyectos mediante la metodología Kanban. En esta aplicación encuentran ciertas limitaciones a la hora de gestionar presupuestos de los proyectos, ver presupuestos generales de cada línea de proyectos, gestionar empleados, proveedores, documentos internos, visualizar información de manera gráfica, hacer diagramas. Debido a estas limitaciones, tienen que usar aplicaciones externas aparte de la propia aplicación web.

El objetivo de este proyecto es proporcionar una aplicación funcional que resuelva las carencias encontradas en las aplicaciones que utilizan actualmente, adaptándola a su forma de trabajo.

1.1. Actores

1.1.1 Autoría

Este trabajo fin de grado de Ingeniería Informática ha sido realizado por Yacy Lodos Saranova, en la Universidad de La Rioja.

1.1.2. Tutores

En este trabajo fin de grado actúan como tutor de la empresa José Luis Moreno de Castro, Cristina Serrano como verificadora de la aplicación en cada uno de los sprints y como tutor de la universidad Jesús María Aransay Azofra.

1.1.3 Clientes

Los clientes de esta aplicación serán los empleados de Ecoembes. Una vez acabado el proyecto utilizarán la aplicación diariamente para gestionar sus proyectos.

1.2. Análisis de requisitos

Tras la primera reunión con el tutor de la empresa redacté los requisitos iniciales (*Acta del día 04 de febrero de 2019*) y posteriormente en otra reunión los validamos (*Acta del día 6 de febrero de 2019*).

1.2.1. Requisitos funcionales

La aplicación deberá seguir el proceso de gestión de proyectos utilizado por Ecoembes.

Los proyectos deberán estar divididos por líneas, las cuales contienen áreas temáticas (fijadas por el cliente). Las principales líneas de estos proyectos son: Innovación, Estudios operativos, Formación e Incentivos.

Estas líneas contienen diversas áreas temáticas:

- Innovación: smartwaste, ecodiseño, ciudadano, emprendimiento, general, proyectos europeos. Estas áreas temáticas tienen diversos programas:
 - Smartwaste: plataformaSW, Planta4.0, Caracterizodromo.
 - Ecodiseño: Materiales, Ecodiseño, 2aVida.
 - Ciudadano: ciudadano.
 - Emprendimiento: Lab, MiniLab, Aceleración.
 - General: ACV, Ecoembes Innova, Otros.
 - Proyectos europeos: circpack, plasticircle, realpack, no_presentados.
- Estudios operativos: laboratorio, observatorio, estudios OT.
- Formación: formación 3os.
- Incentivos: incentivos.

Cada línea debe tener asignado un tablero, en el cual debe poder verse por columnas las diversas áreas temáticas y por filas los diversos estados de un proyecto.

Los empleados registrados en la aplicación deben poder registrar un nuevo proyecto/idea inicial. El estado de dicho proyecto debe ser *a evaluar*. Tras evaluar la idea, si el resultado es positivo el empleado debe poder asignar un presupuesto inicial (presupuesto máximo para el proyecto) y el estado del proyecto debe pasar a *en definición* y si es negativo el estado del proyecto debe ser *descartado*.

Tras asignar un presupuesto inicial, se debe aprobar. Para ello los empleados de Ecoembes deben rellenar un documento interno (llamado A3), en el cual se expone la idea del proyecto y la planificación inicial. La aplicación debe permitir a los empleados rellenar dicho documento y descargarlo en pdf.

Tras aprobar el presupuesto inicial el estado del proyecto debe pasar a estado *en curso*. En este estado la aplicación debe permitir modificar el estado de un proyecto, los cuales pueden ser:

- En curso.
- Stand by.
- Finalizado.
- Descartado.

Cada proyecto tendrá un tablero Kanban, en el cual los usuarios registrados puedan ir registrando las tareas de cada proyecto e ir viendo en qué estado están actualmente, estos estados pueden ser:

- To do (hacer).
- Doing (haciendo).
- Done (hecho).

La aplicación debe permitir gestionar los presupuestos de cada uno de los proyectos. Para ello, además de asignar presupuestos se debe poder rellenar, editar y descargar el documento de

adjudicaciones¹, añadir y eliminar facturaciones² y rellenar y descargar la ampliación³ del presupuesto.

La aplicación también debe permitir gestionar documentos internos: notas internas⁴ y documentos de proveedores. Notas internas: se debe permitir rellenarlas, editarlas y descargarlas. Los documentos de los proveedores deben poder almacenarse tras rellenarlos, por lo que en la aplicación se deben poder subir, descargar y eliminar.

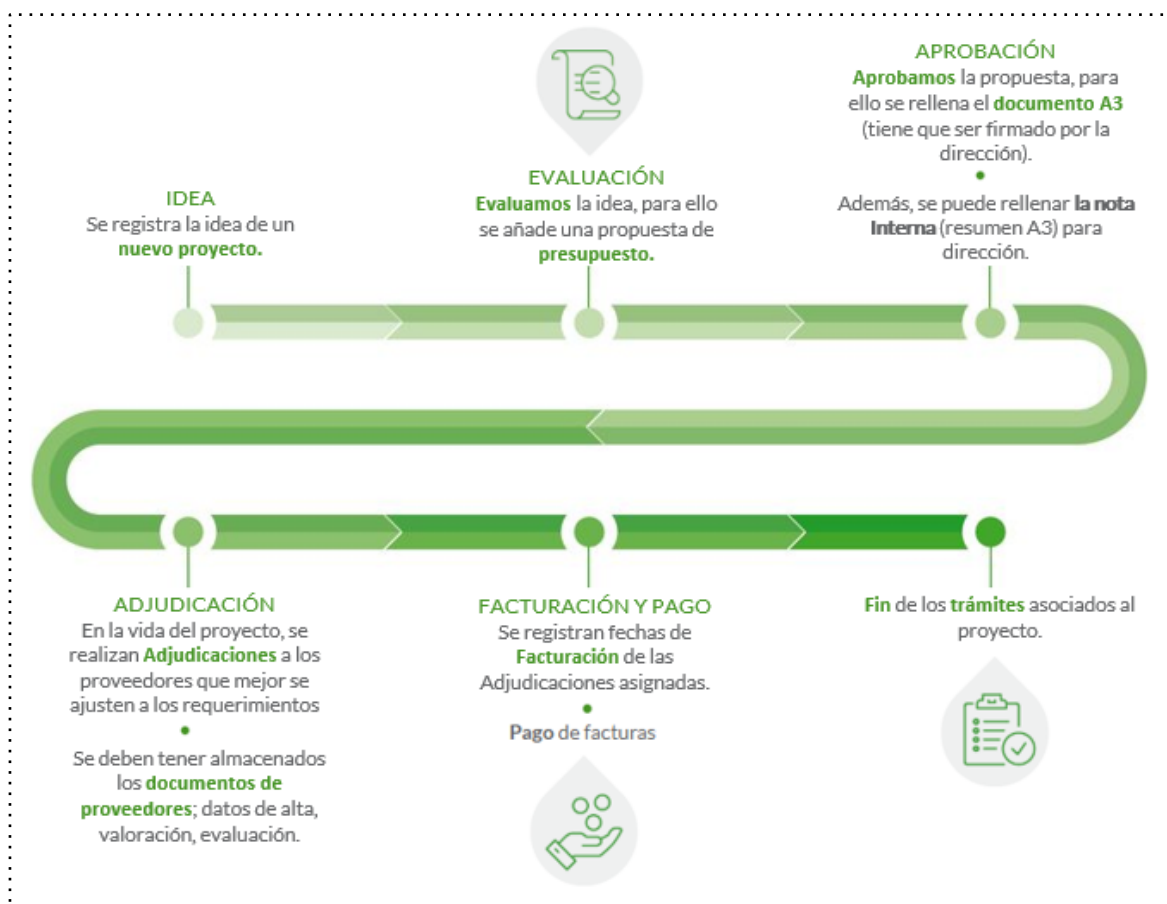


Ilustración 1: proceso de documentación de un proyecto.

¹ **Adjudicaciones:** asignación de una parte del presupuesto del proyecto a un proveedor, para lo cual hay que rellenar un documento/contrato con el proveedor el cual debe poderse rellenar en la aplicación.

² **Facturaciones:** cada una de las adjudicaciones realizadas, se suele pagar en plazos (facturación), tras realizar la adjudicación se especifica fechas, cantidad y en cuantos plazos se pagará la adjudicación. Por lo que tendremos una parte del presupuesto adjudicado (cantidad que hay que pagar a un proveedor) y otra facturado (pagado).

³ **Ampliación del presupuesto:** se estima el presupuesto máximo, pero en ocasiones necesitamos más de lo que hemos estimado. Para ello se debe poder realizar ampliaciones del presupuesto inicial. Se realizará como una adjudicación nueva, pero se debe llamar ampliación del presupuesto para poder diferenciarlas

⁴ **Nota interna:** documento que resume la aprobación del presupuesto (A3)

La aplicación debe permitir a los usuarios modificar sus datos personales y semanalmente debe permitir que documenten su dedicación de tiempo en:

- Innovación
- Incentivos
- Visitas
- Presentaciones
- Formación
- Estudios operativos

Por otro lado, habrá un panel donde poder visualizar datos anuales mediante gráficos:

- Para cada línea de proyecto, se debe poder ver un gráfico dividido por meses donde se compare cantidad adjudicada contra cantidad facturada.
- Porcentaje del presupuesto adjudicado y facturado de cada línea.
- Para cada área temática de la línea de innovación, se debe poder ver un gráfico dividido por meses donde se compare cantidad adjudicada contra cantidad facturada.
- Para cada Empleado, el porcentaje semanal de la dedicación de su tiempo.
- De todos los empleados como conjunto, el porcentaje de dedicación semanal.
- Del porcentaje de proyectos que hay en cada estado.

1.2.2. Requisitos no funcionales

Según las indicaciones del cliente, la aplicación deberá desarrollarse utilizando una metodología de trabajo ágil, con marco de trabajo SCRUM. El código se desarrollará en lenguaje PHP y MySQL, con ayuda del motor de plantillas para php, llamado Smarty.

1.3. Análisis de Mercado

Primero estudiaré si en el mercado actual hay alguna aplicación existente que cubra las necesidades expuestas por el cliente. Para ello investigaré las aplicaciones más utilizadas en la gestión de proyectos. Entre las aplicaciones estudiadas están las siguientes:

- *Trello* [2]: software de administración de gestión de proyectos. Se basa principalmente en tableros Kanban con tarjetas. Esta aplicación no permite gestionar los presupuestos, ni seguir el proceso de trabajo que requiere el cliente.
- *Jira* [3]: herramienta de gestión de proyectos para equipos ágiles. Permite uso de tableros Kanban, Scrum, tiene plantillas para realización de informes y hojas de ruta. Pero como decía antes, esta aplicación tampoco permite gestionar presupuestos.
- *Wrike* [4]: software de colaboración y gestión de proyectos basado en la nube que se adapta a los diferentes equipos de cualquier empresa. Permite creación de diagramas Gantt, gestión de recursos y de cargas de trabajo, colaboración entre equipos, informa del estado en tiempo real, permite realizar informes y paneles de control visuales. Esta aplicación en comparación con las expuestas anteriormente tiene más posibilidades de gestión de un proyecto, pero al igual que en las demás falta el poder gestionar un presupuesto de manera íntegra.

Tras realizar el estudio de mercado buscando la funcionalidad deseada, no se ha encontrado ninguna aplicación que satisfaga todos los requisitos solicitados. El objetivo de mi TFG es desarrollar una aplicación web que permita gestionar proyectos con metodología Kanban y que

además satisfaga los requisitos antes enumerados y otras funcionalidades adicionales que puedan ser requeridas por la empresa.

1.4. Metodología

La metodología que voy a usar va a ser una metodología ágil [5], con un marco de trabajo SCRUM (requisito no funcional del cliente).

La metodología ágil se formula sobre la necesidad de adaptación continua, no suele haber productos finales, sino productos en evolución y mejora. En dicha metodología ágil encontramos el *manifiesto ágil*, un documento que se resume en cuatro valores y doce principios de buenas prácticas para el desarrollo de software.

El manifiesto ágil hace énfasis en *cuatro valores principales*:

1. Los individuos e interacciones por encima de los procesos y las herramientas.
2. Software funcionando por encima de la documentación.
3. La colaboración del cliente por encima de la negociación del contrato.
4. La respuesta al cambio por encima del seguimiento de un plan.

Hemos elegido dicha metodología ya que nos permite ir adaptándonos a las necesidades del cliente e ir probando la funcionalidad del producto, aunque no sea el producto final.

Dentro de esta metodología tenemos diversos marcos de trabajo, nosotros usaremos el marco Scrum. Este marco consta de: el equipo Scrum y sus roles, eventos, artefactos y reglas asociadas. De todas las prácticas que contempla el marco Scrum, pasaré a presentar las que estarán presentes en mi TFG.

El Scrum contiene diversos roles para el equipo Scrum; el Product Owner, Development Team y el Scrum Master.

Product Owner, es el dueño del producto y la única persona responsable de gestionar la lista del producto (Product Backlog).

Development Team, profesionales que realizan un incremento del producto terminado. Son autoorganizados, multifuncionales y cada miembro puede tener habilidades específicas, pero la responsabilidad es del equipo en conjunto.

Scrum Master, responsable de ayudar a todos a entender las prácticas y reglas de la cual se compone el Scrum.

El evento de mayor importancia en el Scrum es el **Sprint**, bloque de tiempo de mínimo una semana y máximo 1 mes, en el cual se crea un incremento del producto terminado. En un sprint no se pueden realizar cambios que afecten al objetivo del Sprint, la calidad del producto no disminuye y el alcance puede modificarse y renegociarse con el Product Owner y el Equipo de desarrollo.

Los Sprints contienen y consisten en el Sprint Planning, Daily Scrum, Trabajo de desarrollo, Sprint Review, Sprint Retrospective.

Sprint Planning, en él se planifica el trabajo que se va a realizar durante un Sprint y debe responder a las preguntas de ¿Qué puede entregarse en el Incremento resultante del Sprint que comienza? y ¿Cómo se conseguirá hacer el trabajo necesario para entregar el Incremento?

Daily Scrum, reunión interna diaria del equipo de desarrollo, con duración máxima de 15 minutos, la cual mejora la comunicación y la necesidad de otras reuniones.

Sprint Review, reunión que se realiza al final de cada Sprint para inspeccionar el incremento y adaptar el product backlog (si es necesario). Su objetivo es facilitar la retroalimentación de información y fomentar la colaboración. Su resultado es un Product Backlog revisado y las posibles historias de usuario que se pueden hacer en el siguiente Sprint, además se puede hacer ajustar el Product Backlog para enfocarse a nuevas oportunidades.

Sprint Retrospective, reunión para inspeccionar el Equipo Scrum, identificar los elementos que salieron bien y las posibles mejoras, para poder crear un plan de mejoras de trabajo para el siguiente Sprint.

Los artefactos en Scrum son clave para maximizar la transparencia de la información y proporcionar oportunidades para inspección y adaptación. Scrum está compuesto del Product Backlog, Sprint Backlog e Incremento.

Product Backlog, lista ordenada de requisitos del producto. Esta lista nunca está completa, va evolucionando a medida que el producto y el entorno donde se usará el producto lo hacen.

Sprint Backlog, conjunto de elementos de la lista de producto seleccionados para el Sprint y un plan para entregar el incremento del producto y conseguir el objetivo del Sprint.

Incremento, suma de los elementos de la lista de productos completados durante un Sprint y el valor de los incrementos de todos los Sprint anteriores.

En este proyecto estarán presentes los tres artefactos. El Product Backlog se realizará al comienzo del proyecto y se irá modificando a lo largo del mismo si es necesario, el Sprint Backlog se realizará en la planificación de cada uno de los sprints y el incremento se utilizará al final de cada uno de los sprints para ver el avance del producto.

El cliente decidió optar por esta metodología debido a la respuesta rápida a los cambios que proporciona la metodología y para colaborar directamente en el proceso.

En este proyecto los roles se van a distribuir de la siguiente manera:

Product Owner: José Luis Moreno de Castro.

Scrum Master: Yacy Lodos Saranova.

Development Team: Yacy Lodos Saranova.

Aunque esta metodología está pensada en el trabajo en equipo, el equipo de desarrollo se compondrá de una persona ya que es un proyecto el cual tengo que desarrollar de manera autónoma. Por lo que las reuniones que realizaremos serán *Sprint Planning* y *Sprint Review*, ya que las reuniones *Daily Scrum* y *Sprint Retrospective* son reuniones basadas en mejorar el trabajo en equipo y en este proyecto el equipo de desarrollo está compuesto de una persona.

1.5. EDT

En la siguiente figura podemos ver la EDT del proyecto.

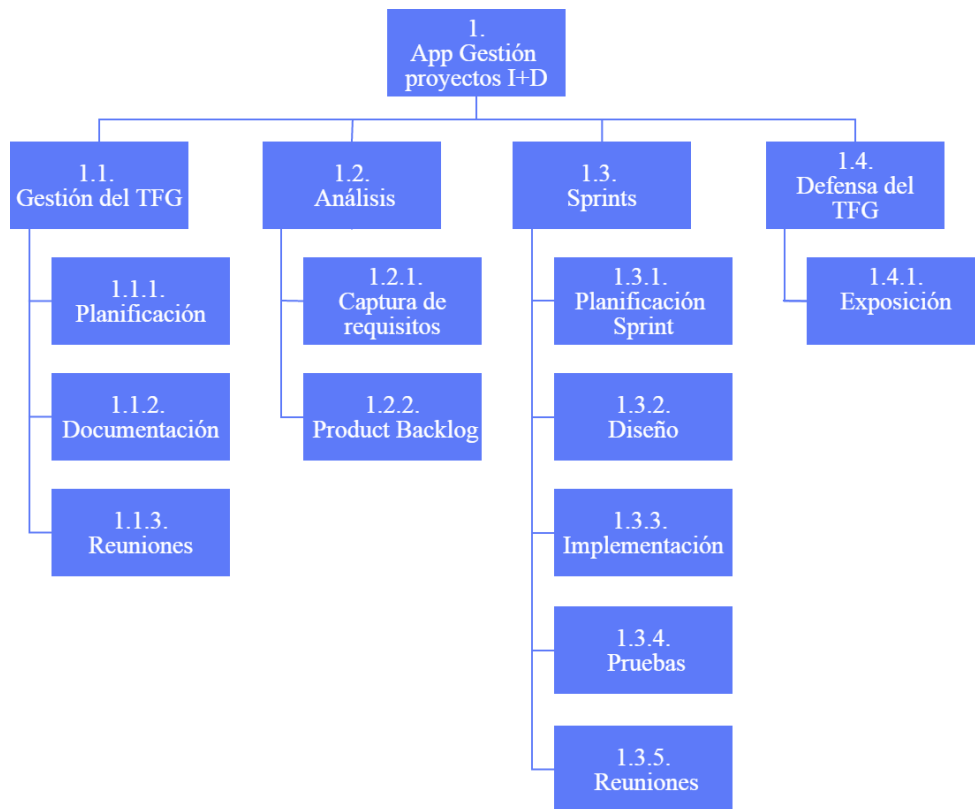


Ilustración 2: Estructura de descomposición de trabajo

1.6. Gantt

Tras la descomposición de trabajo del proyecto, con los diagramas Gantt voy a exponer el tiempo de dedicación previsto a cada tarea del proyecto.

	Febrero 2019													
	S1				S2			S3				S4		
Tareas	4	6	8	10	12	14	16	18	20	22	24	26	28	
1.1.1. Planificación														
1.1.2. Documentación														
1.1.3. Reuniones														
1.2.1. Captura de requisitos														
1.2.2. Product Backlog														
1.3.1. Planificación sprint														
1.3.2. Diseño														
1.3.3. Implementación														
1.3.4. Pruebas														
1.3.5. Reuniones														

Ilustración 2: Diagrama de Gantt mes Febrero

		Marzo 2019															
		S4		S5			S6				S7			S8			
Tareas		1	3	5	7	9	11	13	15	17	19	21	23	25	27	29	31
1.1.2. Documentación																	
1.1.3. Reuniones																	
1.2.1. Captura de requisitos																	
1.2.2. Product Backlog																	
1.3.1. Planificación sprint																	
1.3.2. Diseño																	
1.3.3. Implementación																	
1.3.4. Pruebas																	
1.3.5. Reuniones																	

Ilustración 3: Diagrama de Gantt mes Marzo

	Abril 2019																	
	S9				S10				S11				S12				S13	
Tareas	1	3	5	7	9	11	13	15	17	19	21	23	25	27	29	30		
1.1.2. Documentación																		
1.1.3. Reuniones																		
1.2.1. Captura de requisitos																		
1.2.2. Product Backlog																		
1.3.1. Planificación sprint																		
1.3.2. Diseño																		
1.3.3. Implementación																		
1.3.4. Pruebas																		
1.3.5. Reuniones																		

Ilustración 4: Diagrama de Gantt mes Abril

	Mayo 2019																	
	S13			S14			S15				S16				S17			
Tareas	1	3	5	7	9	11	13	15	17	19	21	23	25	27	29	31		
1.1.1. Planificación																		
1.1.2. Documentación																		
1.1.3. Reuniones																		
1.2.1. Captura de requisitos																		
1.2.2. Product Backlog																		
1.3.1. Planificación sprint																		
1.3.2. Diseño																		
1.3.3. Implementación																		
1.3.4. Pruebas																		
1.3.5. Reuniones																		

Ilustración 5: Diagrama de Gantt mes Mayo

1.7. Product Backlog/ pila de producto

Tras validar los requisitos con el cliente (*Anexo: Acta número 2*), realicé la pila del producto con las historias de usuario, priorizándolas con el método MoSCoW (Must, Should, Could, Won't).

Para la estimación del esfuerzo que conlleva cada historia de usuario, utilicé la variación de la sucesión de Fibonacci (0, ½, 1, 2, 3, 5, 8, 13, 21, infinito) como puntos de historia. En dicha variación se añade el 0 para las historias que no requieren de esfuerzo, ½ para las historias muy pequeñas y el infinito ya que se considera que las historias que pasan de 21 puntos de historia deben dividirse en historias menores.

En mi TFG la estimación de puntos de historia está desde 1 hasta 8 y en cuanto al método MoSCoW no encontraremos historias que estén en priorización *won't/no incluir*, ya que el cliente ha sido concreto en los requisitos de la aplicación.

1.7.1. Imprescindibles/Must have

Requisitos imprescindibles que deben estar en el producto, para que el mismo se pueda considerar un éxito.

Historia de usuario	
Login	2 puntos de historia
Tarjeta	
Como usuario quiero poder iniciar sesión mediante un usuario y una contraseña para poder acceder a la aplicación.	
Criterios de aceptación	
Si el usuario y contraseña es correcto la aplicación nos mostrará la página de inicio de sesión con la cuenta iniciada y si no lo es, nos devolverá un mensaje de error.	

Historia de usuario	
Dedicación de tiempo de usuario	2 puntos de historia
Tarjeta	
Como usuario quiero poder añadir mi dedicación de tiempo semanal para poder tener un registro de dedicación semanal por empleado.	
Criterios de aceptación	
Se mostrará el registro semanal añadido, si hay algún fallo nos devolverá un mensaje de error.	

Historia de usuario	
Añadir proyecto	3 puntos de historia
Tarjeta	
Como usuario quiero poder añadir un nuevo proyecto para poder evaluarlo.	
Criterios de aceptación	
Si se añade correctamente se podrá visualizar en el tablón de la línea correspondiente a la que pertenezca el proyecto añadido, en el estado a evaluar. Si se produce algún fallo nos devolverá un mensaje de error.	

Historia de usuario	
Añadir presupuesto	3 puntos de historia
Tarjeta	
Como usuario quiero poder asignar un presupuesto a un proyecto para gestionar los presupuestos en la aplicación.	
Criterios de aceptación	
Si se añade correctamente el proyecto pasará del estado <i>a evaluar</i> a <i>en definición</i> y se podrá observar en el tablero de la línea. Si no nos devolverá un mensaje de error.	

Historia de usuario	
Añadir ampliación del presupuesto	3 puntos de historia
Tarjeta	
Como usuario quiero poder realizar una ampliación del presupuesto para gestionar los presupuestos de los proyectos.	
Criterios de aceptación	
Si se añade correctamente la aplicación actualizará el presupuesto de un proyecto. Si no nos devolverá un mensaje de error.	

Historia de usuario	
Aprobar presupuesto.	3 puntos de historia
Tarjeta	
Como usuario quiero poder aprobar un presupuesto (rellenar A3) para poder poner en <i>“en curso”</i> el proyecto.	
Criterios de aceptación	
Si se añade correctamente se descargará el A3 (documento interno de la empresa para aprobar un presupuesto) y el proyecto pasará al estado <i>en curso</i> . Si no mostrará un mensaje de error.	

Historia de usuario	
Líneas de proyectos	1 punto de historia
Tarjeta	
Como usuario quiero poder tener los proyectos divididos por líneas para tenerlos organizados.	
Criterios de aceptación	
Tendremos un panel con las distintas líneas de proyectos.	

Historia de usuario	
Tablero línea proyectos	8 puntos de historia
Tarjeta	
Como usuario quiero tener un tablero por cada línea de proyecto, dividido por las áreas temáticas de dicha línea y por los estados del proyecto.	
Criterios de aceptación	
La aplicación mostrará el tablero con los proyectos en sus áreas temáticas y el estado en el que estan.	

Historia de usuario	
Modificar estado de un proyecto	1 puntos de historia
Tarjeta	
Como usuario quiero poder modificar el estado de un proyecto para poder plasmar su estado actual.	
Criterios de aceptación	
La aplicación mostrará el tablero con el estado actual de cada proyecto.	

Historia de usuario	
Tablero Kanban	3 puntos de historia
Tarjeta	
Como usuario quiero poder tener un tablero Kanban dentro de cada proyecto para poder ver las tareas de dicho proyecto.	
Criterios de aceptación	
La aplicación nos mostrará el tablero Kanban con las tareas existentes en sus estados actuales.	

Historia de usuario	
Añadir tareas al tablero Kanban	2 puntos de historia
Tarjeta	
Como usuario quiero poder añadir una nueva tarea al tablero para poder ver las tareas actuales de un proyecto	
Criterios de aceptación	
Si se añade correctamente la aplicación nos mostrará el tablero Kanban actualizado. Si no nos mostrará un mensaje de error.	

Historia de usuario	
Modificar tareas tablero Kanban	2 puntos de historia
Tarjeta	
Como usuario quiero poder modificar las tareas del tablero para corregir posibles errores o para cambiar el estado de la tarea.	
Criterios de aceptación	
Si se modifica correctamente la aplicación nos mostrará el tablero Kanban actualizado. Si no nos devolverá un mensaje de error.	

Historia de usuario	
Añadir Nota Interna	2 puntos de historia
Tarjeta	
Como usuario quiero poder rellenar una nota interna para poder gestionar los documentos internos de la empresa en la aplicación.	
Criterios de aceptación	
La aplicación almacenará los datos en la base de datos y permitirá descargar el documento rellenado en formato word.	

Historia de usuario	
Modificar Nota Interna	1 puntos de historia
Tarjeta	
Como usuario quiero poder modificar una nota interna para poder gestionar los documentos internos de la empresa en la aplicación.	
Criterios de aceptación	
La aplicación almacenará los nuevos datos en la base de datos y permitirá descargar el documento rellenado en formato word.	

Historia de usuario	
Añadir adjudicación	3 puntos de historia
Tarjeta	
Como usuario quiero poder realizar adjudicaciones de un proyecto para poder gestionar el presupuesto.	
Criterios de aceptación	
La aplicación descarga la adjudicación rellena en pdf y nos permitirá ver un registro de las adjudicaciones de un proyecto.	

Historia de usuario	
Modificar adjudicación	2 puntos de historia
Tarjeta	
Como usuario quiero modificar las adjudicaciones de un proyecto para poder modificar los datos desactualizados o erróneos.	
Criterios de aceptación	
La aplicación modificará los datos de la adjudicación y nos mostrará los datos actualizados.	

Historia de usuario	
Añadir facturaciones.	2 puntos de historia
Tarjeta	
Como usuario quiero poder introducir las fechas de las facturaciones de un proyecto para gestionar el presupuesto.	
Criterios de aceptación	
Si se añade correctamente la aplicación nos mostrará las fechas introducidas en las cuales se realizarán las facturaciones. Si no nos devolverá un mensaje de error.	

Historia de usuario	
Eliminar facturaciones.	1 puntos de historia
Tarjeta	
Como usuario quiero poder eliminar las fechas de las facturaciones de un proyecto para tenerlas actualizadas.	
Criterios de aceptación	
Si se elimina correctamente las fechas eliminadas no se verán en la lista de fechas de adjudicación de un proyecto. Si no devolverá un mensaje de error.	

Historia de usuario	
Subir documento proveedores.	1 puntos de historia
Tarjeta	
Como usuario quiero poder subir los documentos rellenos por un proveedor a la aplicación para tener todos los documentos del proyecto en la aplicación.	
Criterios de aceptación	
Al subirse un documento, se mostrará en el listado de documentos de proveedores.	

Historia de usuario	
Descargar documento proveedores.	1 puntos de historia
Tarjeta	
Como usuario quiero poder descargar el documento de un proveedor para en caso de que lo necesite pueda obtenerlo.	
Criterios de aceptación	
Al pinchar sobre el botón descargar, el documento se descargará o en caso contrario se nos mostrará un mensaje de error.	

Historia de usuario	
Eliminar documentos proveedores.	1 puntos de historia
Tarjeta	
Como usuario quiero poder eliminar los documentos subidos para tener en la aplicación los documentos correctos.	
Criterios de aceptación	
Si se modifican correctamente la aplicación nos mostrará las fechas actualizadas en las cuales se realizarán las facturaciones. Si no nos devolverá un mensaje de error.	

1.7.2. *Importantes/should have*

Requisitos importantes que deberían añadirse al producto final, aunque si hubiera alguna causa justificada podrían ser prescindibles.

Historia de usuario	
Cambio de contraseña	2 puntos de historia
Tarjeta	
Como usuario quiero poder cambiar mi contraseña para que en caso de olvido pueda acceder a la aplicación.	
Criterios de aceptación	
Si se modifica correctamente la aplicación nos proporcionará un mensaje de confirmación. Si no nos devolverá un mensaje de error.	

Historia de usuario	
Descargar Aprobación de un presupuesto (documento A3).	5 puntos de historia
Tarjeta	
Como usuario quiero poder descargar el documento A3 tras rellenarlo para que pueda ser aprobado.	
Criterios de aceptación	
La aplicación descargará en formato pdf el documento A3 rellenado.	

Historia de usuario	
Descargar adjudicaciones.	3 puntos de historia
Tarjeta	
Como usuario tras realizar una adjudicación quiero poder tener la opción de descargar la plantilla de adjudicación rellenada.	
Criterios de aceptación	
La aplicación descargará la adjudicación en formato word.	

Historia de usuario	
Ver características de un proyecto	1 punto de historia
Tarjeta	
Como usuario quiero poder ver las características de un proyecto para poder saber en qué consiste el proyecto.	
Criterios de aceptación	
La aplicación nos mostrará una ventana con los detalles más relevantes de un proyecto.	

Historia de usuario	
Eliminar tareas al tablero Kanban	2 puntos de historia
Tarjeta	
Como usuario quiero poder eliminar tareas del tablero Kanban para poder tener el tablero actualizado.	
Criterios de aceptación	
Si se elimina correctamente la tarea se eliminará de la base de datos y la aplicación nos mostrará el tablero Kanban actualizado. Si no nos mostrará un mensaje de error.	

Historia de usuario	
Gráfico de presupuesto por líneas.	5 puntos de historia
Tarjeta	
Como usuario quiero visualizar un gráfico por cada línea de sus datos anuales, de las adjudicaciones y facturaciones mensuales con la constante del presupuesto anual.	
Criterios de aceptación	
La aplicación nos mostrará los datos actualizados en el gráfico.	

Historia de usuario	
Gráfico de presupuesto por áreas temáticas de innovación.	5 puntos de historia
Tarjeta	
Como usuario quiero poder tener un gráfico por cada área temática de la línea de innovación anual, para visualizar las adjudicaciones y facturaciones mensuales con la constante del presupuesto anual.	
Criterios de aceptación	
La aplicación mostrará el gráfico actualizado.	

Historia de usuario	
Gráfico dedicación empleados	5 puntos de historia
Tarjeta	
Como usuario quiero poder tener un gráfico de la dedicación de horas semanales (%) de cada usuario para poder tener registrada la actividad de cada usuario.	
Criterios de aceptación	
La aplicación mostrará el gráfico actualizado semanalmente.	

Historia de usuario	
Gráfico porcentaje estados de los proyectos	2 puntos de historia
Tarjeta	
Como usuario quiero poder tener un gráfico del porcentaje de los estados de los proyectos para ver el porcentaje de proyectos que hay, en los estados de un proyecto.	
Criterios de aceptación	
La aplicación mostrará el gráfico actualizado.	

Historia de usuario	
Gráfico porcentaje de dedicación de todos los empleados.	3 puntos de historia
Tarjeta	
Como usuario quiero poder ver un gráfico del porcentaje de dedicación del tiempo de todos los empleados en las diversas actividades.	
Criterios de aceptación	
La aplicación mostrará el gráfico actualizado.	

Historia de usuario	
Gráfico porcentaje total facturado y adjudicado sobre un presupuesto anual	2 puntos de historia
Tarjeta	
Como usuario quiero poder ver un gráfico del porcentaje facturado y adjudicado sobre una línea para poder ver las estadísticas.	
Criterios de aceptación	
La aplicación mostrará el gráfico actualizado.	

1.7.3. Buenas/could have

Requisitos deseables, pero no necesarios para el éxito del producto.

Historia de usuario	
Editar perfil	2 puntos de historia
Tarjeta	
Como usuario quiero poder editar mi perfil para tener mis datos personales actualizados.	
Criterios de aceptación	
Si se modifica correctamente la aplicación nos mostrará el perfil actualizado. Si no nos mostrará un mensaje de error.	

Historia de usuario	
Añadir Proveedor	2 puntos de historia
Tarjeta	
Como usuario quiero poder añadir un proveedor para tener añadidos en la aplicación los proveedores con los que trabajamos	
Criterios de aceptación	
Si se añade correctamente el proveedor nos aparecerá un mensaje de confirmación.	

1.8. Entregables

Al acabar el proyecto tendré dos principales entregables; uno será la memoria la cual recogerá toda la planificación y todo el proceso que he seguido para desarrollar la aplicación. Por otro lado, también tendré la aplicación, la cual estará desplegada en el servidor de pruebas que me proporcione la empresa.

1.9. Exclusiones

Este proyecto está destinado a la hacer una aplicación básica pero funcional, debido al escaso tiempo. Por lo que se descarta todo tipo de funcionalidades que no sean las mínimas para el buen funcionamiento de la aplicación. Entre ellas podemos destacar:

Parte de administrador la cual se deberá poder:

- añadir y eliminar empleados
- añadir, eliminar y editar presupuestos anuales.
- eliminar, editar y añadir proveedores.
- Borrar proyectos.
- Añadir, eliminar y editar departamentos.

Por otro lado, no se espera hacer documentación/tutorial de uso, ya que no ha sido requerido por el tutor de la empresa, ni el soporte y mantenimiento de esta posteriormente a la finalización del proyecto.

2. Planificación

La planificación del sprint tiene que constar de qué le podemos entregar al cliente tras el sprint y como se va a conseguir ese incremento.

La duración de los sprints será de tres semanas e intentaré que estén compuestos de 17 o 18 puntos de historia. Por lo que, si yo hago 18 horas semanales, cada sprint será de 54 horas, las cuales la dedicaré al incremento de la aplicación, a la documentación de la memoria, a posibles reuniones y a posibles desviaciones.

Para la elección de las historias de usuario de cada sprint he ido siguiendo el orden del proceso de creación de un proyecto y como actúa el usuario en dicho proceso.

2.1. Sprint 0

Este sprint no va a ser un sprint como tal, ya que la duración será de dos semanas, las cuales las voy a dedicar a formación y a planificación. Esto no le aporta valor al producto, no es un incremento para el cliente.

► febrero 2019

lu.	ma.	mi.	ju.	vi.	sá.	do.
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28			

En él recojo los requisitos de la aplicación (Anexo: Acta de reunión número 1), los valido con el cliente (Anexo: Acta de reunión número 2), genero la pila de producto y realizo un curso sobre la metodología escogida por el cliente (metodología ágil, con marco SCRUM).

2.2. Sprint 1

Este Sprint durará desde el 18 de febrero hasta el 10 marzo. Tras la formación y la planificación previa, en este sprint comenzaré con el primer incremento de la aplicación.

► febrero 2019

► marzo 2019

lu.	ma.	mi.	ju.	vi.	sá.	do.	lu.	ma.	mi.	ju.	vi.	sá.	do.
				1	2	3					1	2	3
4	5	6	7	8	9	10	4	5	6	7	8	9	10
11	12	13	14	15	16	17	11	12	13	14	15	16	17
18	19	20	21	22	23	24	18	19	20	21	22	23	24
25	26	27	28				25	26	27	28	29	30	31

2.2.1. Sprint Backlog

2.2.1.1. Historias de Usuario.

El primer Sprint está compuesto de las siguientes historias de usuario y la estimación de todo el sprint es *18 puntos de historia*. Esta estimación intentaré que sea igual o parecida en los siguiente sprints para que sean equitativos.

Prioridad	Historia de Usuario	Estimación	Responsable
1	Login	2	Yacy Lodos Saranova
2	Añadir proyecto	3	
3	Líneas de proyectos	1	
4	Tablero Líneas proyectos	8	
5	Añadir presupuesto	3	
6	Ver características de un proyecto	1	

Tabla 1: Historias de usuario sprint 1

2.2.1.2. Descomposición de historias de usuario en tareas.

Proseguimos a descomponer en tareas las historias de usuario anteriores.

Historia de Usuario	Tarea	Descripción
Login	Tarea 1	Diseño de la tabla "Empleado" en la base de datos.
	Tarea 2	Implementación de los métodos de la base de datos.
	Tarea 3	Diseño de la interfaz.
	Tarea 4	Funcionalidad de la interfaz.
	Tarea 5	Pruebas de funcionalidad.
Añadir proyecto	Tarea 6	Diseño de la tabla "Proyecto" en la base de datos.
	Tarea 7	Implementación de los métodos de la base de datos.
	Tarea 8	Diseño y funcionalidad del formulario para añadir proyecto.
	Tarea 9	Pruebas funcionales.
Líneas de proyectos	Tarea 10	Diseño de la interfaz.
	Tarea 11	Funcionalidad de la interfaz.
Tablero líneas proyecto	Tarea 12	Diseño de la interfaz del tablero.
	Tarea 13	Funcionalidad del tablero.

Tabla 2: Tareas sprint 1

Historia de Usuario	Tarea	Descripción
Añadir presupuesto	Tarea 14	Diseño de la tabla "Presupuesto" en la base de datos.
	Tarea 15	Implementación de los métodos de la base de datos.
	Tarea 16	Diseño y funcionalidad del formulario para añadir presupuesto.
	Tarea 17	Pruebas funcionales.
Ver características de un proyecto	Tarea 18	Diseño y funcionalidad de la interfaz.

Tabla 3: Tareas sprint 1

2.3. Sprint 2

Este Sprint durará desde el 11 de marzo hasta el 31 de marzo. En el seguiremos con el incremento de la aplicación.

▶ marzo 2019

lu.	ma	mi.	ju.	vi.	sá.	do.
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

2.3.1. Sprint Backlog

2.3.1.1. Historias de Usuario.

El Segundo Sprint está compuesto de las siguientes historias de usuario y la estimación está en 18 puntos de historia.

Prioridad	Historia de Usuario	Estimación	Responsable
1	Aprobar presupuesto	3	Yacy Lodos Saranova
2	Modificar estado proyecto	1	
3	Descargar A3	5	
4	Tablero Kanban	3	
5	Añadir Tareas Kanban	2	
6	Modificar tareas Kanban	2	
7	Eliminar tareas Kanban	2	

Tabla 4: Historias de usuario sprint 2.

2.3.1.2. Descomposición de historias de usuario en tareas.

Proseguimos a descomponer en tareas las historias de usuario anteriores.

Historia de Usuario	Tarea	Descripción
Aprobar presupuesto	Tarea 1	Implementar las funciones necesarias para añadir los nuevos datos a la tabla "Proyecto"
	Tarea 2	Diseñar e implementar la funcionalidad de la interfaz necesaria para este proceso
	Tarea 3	Pruebas de funcionalidad
Modificar estado proyecto	Tarea 4	Implementar la función necesaria para el cambio de estado en la base de datos
	Tarea 5	Añadir un botón en la tarjeta del proyecto que permita el cambio.
	Tarea 6	Pruebas del correcto funcionamiento
Descargar A3	Tarea 7	Buscar librería que me permita realizar la descargar en formato pdf
	Tarea 8	Implementar la funcionalidad de la descarga.
	Tarea 9	Pruebas de funcionalidad
Tablero Kanban	Tarea 10	Diseñar e implementar la interfaz
Añadir tareas Kanban	Tarea 11	Diseñar e implementar la tabla "Tareas"
	Tarea 12	Implementar de los métodos de la base de datos
	Tarea 13	Diseñar e implementar la interfaz para esta funcionalidad
	Tarea 14	Pruebas de funcionalidad
Modificar tareas Kanban	Tarea 15	Implementar la funcionalidad para modificar una tarea
	Tarea 16	Pruebas de funcionalidad
Eliminar tareas Kanban	Tarea 17	Implementar función para eliminar una tarea
	Tarea 18	Pruebas de funcionalidad

Tabla 5: Tareas sprint 2

2.4. Sprint 3

Este sprint durará desde el 1 de abril hasta el 21 de abril.

▶ abril 2019

lu.	ma	mi.	ju.	vi.	sá.	do.
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

2.4.1. Sprint Backlog

2.4.1.1. Historias de Usuario.

El tercer Sprint está compuesto de las siguientes historias de usuario y la estimación está en 17 puntos de historia.

Prioridad	Historia de Usuario	Estimación	Responsable
1	Añadir Adjudicación	3	Yacy Lodos Saranova
2	Modificar Adjudicación	2	
3	Descargar Adjudicación	3	
4	Añadir nota interna	2	
5	Modificar nota interna	1	
6	Subir documento proveedores	1	
7	Descargar documento proveedores	1	
8	Eliminar documento proveedores	1	
9	Añadir ampliación presupuesto	3	

Tabla 6: Historias de usuario sprint 3.

2.4.1.2. Descomposición de historias de usuario en tareas.

Historia de Usuario	Tarea	Descripción
Añadir Adjudicación	Tarea 1	Diseñar la tabla “Adjudicacion”.
	Tarea 2	Implementar las funciones necesarias para añadir los nuevos datos a la tabla “Adjudicacion”
	Tarea 3	Diseñar e implementar la funcionalidad del formulario.
	Tarea 4	Pruebas de funcionalidad.
Modificar Adjudicación	Tarea 5	Implementar la funcionalidad para reescribir los datos alojados en la base de datos.
	Tarea 6	Pruebas del correcto funcionamiento.
Descargar Adjudicación	Tarea 7	Buscar y estudiar librería para realizar la descarga en formato word.
	Tarea 8	Implementar la funcionalidad de la descarga.
Añadir Nota Interna	Tarea 9	Diseñar la tabla “notaInterna”.
	Tarea 10	Implementar las funciones necesarias para añadir los nuevos datos a la tabla “notaInterna”
	Tarea 11	Diseñar e implementar la funcionalidad del formulario.
	Tarea 12	Pruebas de funcionalidad.
Modificar Nota Interna	Tarea 13	Implementar la funcionalidad para reescribir los datos alojados en la base de datos.
	Tarea 14	Pruebas de funcionalidad.
Subir documento proveedores	Tarea 15	Implementar funcionalidad de la subida de documentos al servidor.
	Tarea 16	Diseñar e implementar interfaz para listar los ficheros de un proyecto
	Tarea 17	Pruebas de funcionalidad.
Descargar documento proveedores.	Tarea 18	Implementar funcionalidad de la descarga de documentos del servidor.
	Tarea 19	Pruebas de funcionalidad.

Tabla 7: Tareas sprint 3

Historia de Usuario	Tarea	Descripción
Eliminar documento proveedores	Tarea 20	Implementar funcionalidad de eliminación de un fichero.
	Tarea 21	Pruebas de funcionalidad
Añadir ampliación presupuesto	Tarea 22	Añadir una nueva columna llamada ampliación en la tabla "Adjudicación".
	Tarea 23	Implementar la funcionalidad de ampliación.
	Tarea 24	Pruebas de funcionalidad.

Tabla 8: Tareas sprint 3

2.5. Sprint 4

► abril 2019

► mayo 2019

lu.	ma	mi.	ju.	vi.	sá.	do.	lu.	ma	mi.	ju.	vi.	sá.	do.
1	2	3	4	5	6	7			1	2	3	4	5
8	9	10	11	12	13	14	6	7	8	9	10	11	12
15	16	17	18	19	20	21	13	14	15	16	17	18	19
22	23	24	25	26	27	28	20	21	22	23	24	25	26
29	30						27	28	29	30	31		

2.5.1. Sprint Backlog

2.5.1.1. Historias de Usuario.

El cuarto Sprint está compuesto de las siguientes historias de usuario y la estimación está en 17 puntos de historia.

Prioridad	Historia de Usuario	Estimación	Responsable
1	Añadir facturaciones	2	Yacy Lodos Saranova
2	Eliminar facturaciones	1	
3	Dedicación tiempo usuario	2	
4	Cambio contraseña	2	
5	Gráfico de presupuesto por línea	5	
6	Gráfico de presupuesto áreas temáticas innovación.	5	

Tabla 9: Historias de usuario sprint 4.

2.5.1.2. Descomposición de historias de usuario en tareas.

Historia de Usuario	Tarea	Descripción
Añadir Facturaciones	Tarea 1	Diseñar la tabla "Factura".
	Tarea 2	Implementar las funciones necesarias para añadir los nuevos datos a la tabla "Factura"
	Tarea 3	Diseñar e implementar la funcionalidad del formulario.
	Tarea 4	Pruebas de funcionalidad.
Eliminar Facturaciones	Tarea 5	Implementar la funcionalidad para eliminar los datos de la base de datos
	Tarea 6	Pruebas del correcto funcionamiento.
Dedicación tiempo de usuario	Tarea 7	Añadir tabla "Dedicación"
	Tarea 8	Implementar las funciones necesarias para añadir los nuevos datos a la tabla "Dedicación"
	Tarea 9	Diseñar e implementar la funcionalidad del formulario.
	Tarea 10	Pruebas de funcionalidad.
Cambio de contraseña	Tarea 11	Habilitar una interfaz para cambio de contraseña.
	Tarea 12	Proporcionar funcionalidad a la interfaz
	Tarea 13	Pruebas de funcionalidad.
Gráfico de presupuesto por línea.	Tarea 14	Estudiar los gráficos de google.
	Tarea 15	Añadir métodos que recojan los datos necesarios.
	Tarea 16	Crear un panel para gráficos.
	Tarea 17	pruebas de funcionalidad
Gráfico de presupuesto áreas temáticas innovación.	Tarea 18	Añadir los métodos necesarios para recoger los datos.
	Tarea 19	Añadir gráfico al panel de gráficos.
	Tarea 20	Pruebas de funcionalidad.

Tabla 10: Tareas sprint 4.

2.6. Sprint 5

► mayo 2019

lu.	ma	mi.	ju.	vi.	sá.	do.
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

2.6.1. Sprint Backlog

2.6.1.1. Historias de Usuario.

El Segundo Sprint está compuesto de las siguientes historias de usuario y la estimación está en 18 puntos de historia.

Prioridad	Historia de Usuario	Estimación	Responsable
1	Gráfico dedicación empleados	5	Yacy Lodos Saranova
2	Gráfico porcentaje estados de los proyectos	2	
3	Gráfico porcentaje de dedicación de todos los empleados	3	
4	Gráfico porcentaje adjudicado y facturado sobre un presupuesto anual	2	
5	Editar perfil	2	
6	Añadir proveedor	2	
7	Posibles cambios/errores	2	

Tabla 11: Historias de usuario sprint 5.

2.6.1.2. Descomposición de historias de usuario en tareas.

Historia de Usuario	Tarea	Descripción
Gráfico dedicación empleados	Tarea 1	Añadir los métodos necesarios para recoger los datos.
	Tarea 2	Añadir gráfico al panel de gráficos.
	Tarea 3	Pruebas de funcionalidad.

Tabla 12: Tareas sprint 5.

Historia de Usuario	Tarea	Descripción
Gráfico porcentaje estados de los proyectos	Tarea 4	Añadir los métodos necesarios para recoger los datos.
	Tarea 5	Añadir gráfico al panel de gráficos.
	Tarea 6	Pruebas de funcionalidad.
Gráfico porcentaje de dedicación de todos los empleados	Tarea 7	Añadir los métodos necesarios para recoger los datos.
	Tarea 8	Añadir gráfico al panel de gráficos.
	Tarea 9	Pruebas de funcionalidad.
Gráfico porcentaje adjudicado y facturado sobre un presupuesto anual	Tarea 10	Añadir los métodos necesarios para recoger los datos.
	Tarea 11	Añadir gráfico al panel de gráficos.
	Tarea 12	Pruebas de funcionalidad.
Editar perfil	Tarea 13	Añadir los métodos necesarios para recoger los datos.
	Tarea 14	Añadir gráfico al panel de gráficos.
	Tarea 15	Pruebas de funcionalidad.
Añadir proveedor	Tarea 16	Diseñar la tabla "Proveedor".
	Tarea 17	Implementar las funciones necesarias.
	Tarea 18	Diseñar e implementar la funcionalidad del formulario.
	Tarea 19	Pruebas de funcionalidad.
Posibles cambios/errores	Tarea 20	Corregir posibles errores o cambios.

Tabla 13: Tareas sprint 5.

3. Sprint

3.1. Sprint 1


3.1.1. Incremento

Tras la realización de este sprint, tendremos los siguientes incrementos:

1. Formulario de registro de un Empleado.
2. Login.
3. Inicio de la aplicación.
4. Formulario para añadir proyectos.
5. Panel de líneas de proyectos.
6. Tablero líneas de proyectos.
7. Tabla para ver las características de un proyecto.

3.1.2. Trabajo de desarrollo

En cuanto al trabajo de desarrollo, lo primero fue diseñar la tabla “*Empleado*” con lo necesario para el Login y pensando en los datos necesarios para los posteriores Sprints.



nombre	: varchar(11)
apellidos	: varchar(45)
puesto	: varchar(45)
email	: varchar(45)
tlf	: varchar(45)
usr	: varchar(11)
pwd	: varchar(45)
palabraClave	: varchar(45)

Tras el diseño de la tabla implemento la clase *Empleado* con los atributos especificados en la base de datos, una clase *GestorBD* para poder gestionar el acceso a la base de datos desde un mismo fichero y la clase *Conect* para realizar las conexiones y desconexiones.

Utilizo php ya que era uno de los requisitos no funcionales pedidos por el cliente. Por ejemplo, la clase *Conect* realizada con php quedaría de la siguiente manera:

```
<?php
class Conect{
    function conectarBD(){
        //datos de conexión
        $server = "";
        $usuario = "";
        $pass = "";
        $BD = "";
        //variable que guarda la conexión de la base de datos
        $conexion = mysqli_connect($server, $usuario, $pass, $BD);
        $conexion->set_charset("utf8");
        return $conexion;
    }

    //Cierra la conexión y guarda el estado de la operación en una variable
    function desconectarBD($conexion){
        $close = mysqli_close($conexion);
        //devuelve el estado del cierre de conexión
        return $close;
    }
}
?>
```

Para realizar el Login, en *GestorBD* introduzco los métodos necesarios para comprobar si un usuario ya está registrado (por nombre y apellidos), verificar usuario y contraseña y listar todos los empleados. Un ejemplo de como comprobar si existe un empleado o de como listar en php, es la siguiente:

```
//nos devuelve true si hay un empleado con dicho nombre y apellidos.
public function empleadoNomAp11( $nombre, $apellidos ) {
    $c = new Conect();
    $con = $c->conectarBD();

    $stmt = $con->prepare("SELECT * FROM empleado where nombre=? and
    apellidos=?");
    $stmt->bind_param("ss", $nombre,$apellidos);

    //ejecutar y obtener
    $stmt->execute();
    $stmt->store_result();

    return $stmt->num_rows;
    $c->desconectarBD( $con );
}

}

//devuelve la lista de empleados.
public function listaEmpleados() {
    $c = new Conect();
    $con = $c->conectarBD();

    $sqlS = "SELECT usr , nombre , apellidos FROM empleado";
    if ( $result = mysqli_query( $con, $sqlS ) ) {
        $i = 0;
        $list = array();
        while ( $row = mysqli_fetch_array( $result ) ) {
            $list[ $i ] = $row[ 0 ] . " , " . $row[ 1 ] . " " . $row[
            2 ];

            $i++;
        }
        return ( $list );
    } else {
        echo "no hay ningún empleado";
    }

    $c->desconectarBD( $con );
}
```

Para el diseño de la interfaz he elegido el framework Bootstrap [6], el cual a diferencia de los demás frameworks solo se basa en el front-end de la aplicación. Contiene elementos de diseño (botones, menús, formularios...) basados en html, css y extensiones javascript y permite realizar una aplicación adaptable a cualquier dispositivo (responsive). Utilizaré este framework durante todo el proyecto, puesto que puedo hacer el diseño rápidamente adaptado al diseño que utiliza la empresa.

Tras realizar un curso [7] sobre dicho framework, lo primero que hice fue diseñar e implementar la funcionalidad de un formulario para poder registrar a un empleado nuevo. Para separar la vista del controlador voy a utilizar Smarty [8]; es un motor de plantillas de php que me ayuda mediante etiquetas a tener la lógica de negocio separada de la presentación, herramienta sencilla con curva de aprendizaje baja.

Tras los anteriores pasos, diseño e implemento la interfaz de Login, página de inicio de sesión de la aplicación. En el controlador compruebo si el usuario existe, si existe inicio sesión y si no devuelvo un mensaje de error.

Realizo pruebas de funcionalidad, para ello compruebo:

- Si un usuario añadido anteriormente accede a la aplicación
- Si tras poner mal la contraseña o mal el usuario me muestra un mensaje de error
- Si se pueden realizar inyecciones de SQL (para lo cual tomé las previsiones de hacer las consultas con prepared statement en php y cifrar las contraseñas con el algoritmo de cifrado SHA).

Seguimos con el diseño de la tabla “*Proyecto*” con lo necesario para almacenar los datos requeridos por el cliente.



nuevabd proyecto	
idproy	varchar(100)
nombre	varchar(100)
responsable	varchar(45)
departamento	varchar(11)
linea	varchar(45)
areatematica	varchar(45)
programa	varchar(45)
viaFinanciacion	varchar(50)
objetivo	varchar(500)
alcance	varchar(500)
beneficiosEstimados	varchar(500)
roi	varchar(45)
proveedores	varchar(45)
fecha_ini	date
fecha_fin	date
hitos_criticos	varchar(500)
estado	varchar(45)
medios_necesarios	varchar(45)
coste	varchar(45)
entregables	varchar(500)
debilidades	varchar(300)
amenazas	varchar(300)
fortalezas	varchar(300)
oportunidades	varchar(300)

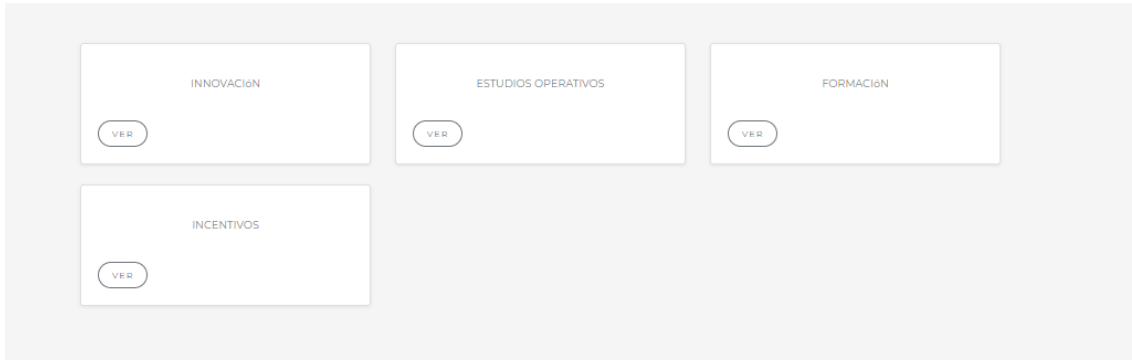
Como vemos en la tabla Proyecto tenemos una clave primaria idProy, la cual nos pidió el cliente que estuviera compuesta de las tres primeras letras de la línea + los dos últimos dígitos del año actual + un número incremental y *nombre* (el cual debe ser único).

Tras el diseño de la tabla seguimos con la implementación de la clase Proyecto y añadimos en GestorBD las funciones necesarias para añadir un proyecto, modificar, listar y devolver un proyecto proporcionando un *idproy* o un *nombre* y *cambiar el estado*.

Diseño e implemento el formulario para poder añadir un proyecto, y un script php para realizar las comprobaciones necesarias, generar el idproy y añadir el proyecto en la base de datos. Tras estos pasos realizo pruebas de funcionalidad, compruebo que se añade un proyecto correctamente, que el idproy requerido por el cliente es correcto y que si hay algún error al añadir un proyecto nos muestra un mensaje de error.

Sigo con la implementación y diseño de la pantalla donde vamos a ver las líneas de proyectos existentes (*ilustración 6*).

Panel de proyectos



© 2018-2019 Ecoembes

Ilustración 6: tablero de proyectos

El tablero líneas de proyectos (*ilustración 7*) va a ser un tablero donde cada columna sea un área temática de la línea que el cliente haya seleccionado. Cada columna estará dividida en los posibles estados del proyecto (*en curso*, *stand by* y *finalizado*).

Además, habrá dos columnas adicionales, una para nuevas propuestas, también dividida en dos posibles estados (*a evaluar* y *en definición*) y otra para propuestas descartadas. Diseño y le doy funcionalidad para que los proyectos añadidos se listen en su correspondiente columna.

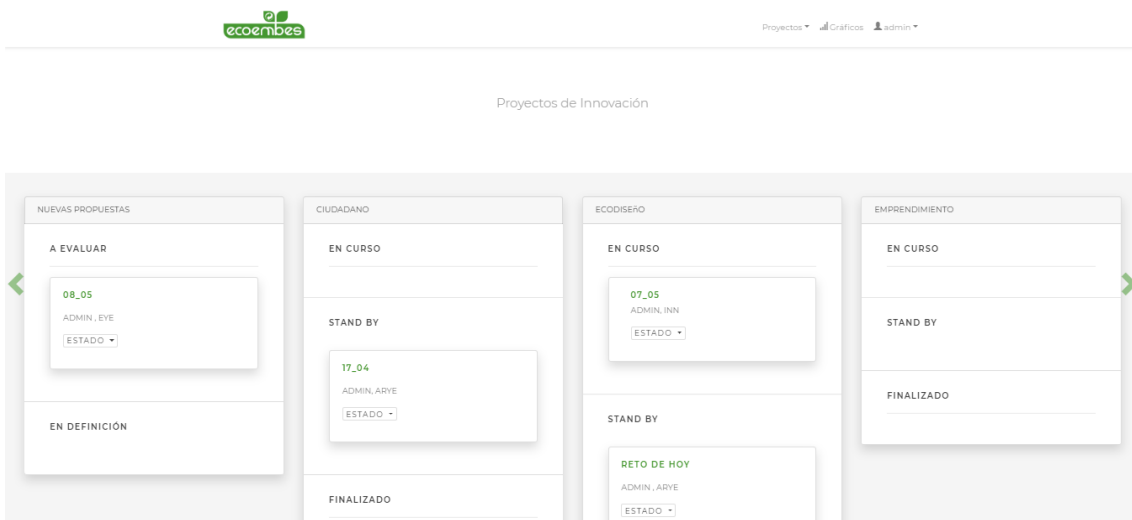
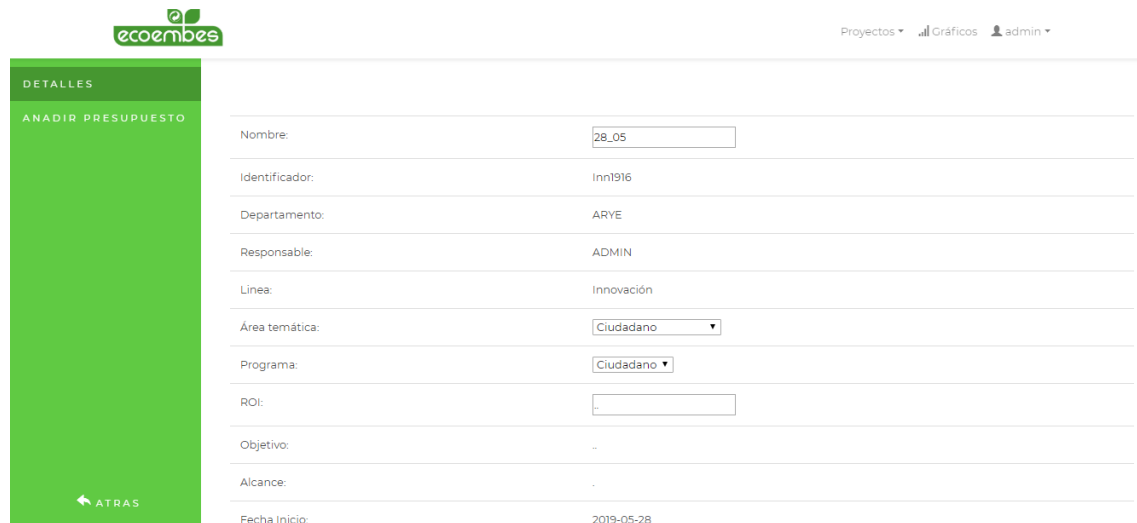


Ilustración 7: Tablero áreas proyectos

Los proyectos recién añadidos tienen estado “a evaluar”. Para que cambie el estado a “en definición” tenemos que añadir un presupuesto inicial. En la tarjeta de los proyectos con estado “a evaluar” al pulsar el nombre del proyecto, nos lleva a una pantalla con un menú vertical con las opciones para ver los detalles del proyecto y con un formulario para añadir el presupuesto inicial (ilustración 8).



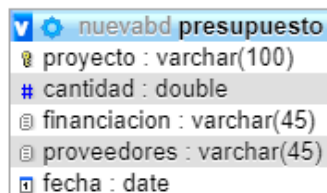
The screenshot shows the 'ecoembes' web application interface. On the left is a green sidebar with 'DETALLES' and 'ANADIR PRESUPUESTO' options. The main area contains a form with the following fields:

Nombre:	28_05
Identificador:	Inn1916
Departamento:	ARYE
Responsable:	ADMIN
Línea:	Innovación
Área temática:	Ciudadano
Programa:	Ciudadano
ROI:	-
Objetivo:	-
Alcance:	-
Fecha Inicio:	2019-05-28

At the bottom of the sidebar is an 'ATRÁS' button with a back arrow icon.

Ilustración 8: Detalles del proyecto diseño final.

Para añadir un presupuesto diseño la tabla “Presupuesto”, en la cual tenemos la clave primaria *proyecto* (un presupuesto tiene un proyecto asignado) y fecha en la que se le asignó el presupuesto.



	nuevabd presupuesto
🔑	proyecto : varchar(100)
#	cantidad : double
📄	financiacion : varchar(45)
📄	proveedores : varchar(45)
📅	fecha : date

Tras realizar el diseño de la tabla, implementamos la clase Presupuesto y añadimos en *GestorBD* métodos para añadir el presupuesto, listar presupuestos de un proyecto y ver si un presupuesto es de un proyecto. Diseño e implemento la funcionalidad del formulario y añado un script para realizar las

comprobaciones necesarias, añadir el presupuesto y cambiar el estado del proyecto a “en definición”.

Realizo pruebas funcionales del tablero de líneas, para ver que cuando añado un nuevo proyecto se lista en su correspondiente tablero, que a dicho proyecto le puedo asignar un presupuesto, que se añade adecuadamente en la base de datos, que dicho proyecto cambia de estado y se lista correctamente. También compruebo que en el apartado detalles se listen los datos correctamente.

3.1.3. Sprint Review

Se realiza una reunión para validar lo realizado en el sprint 1 y se recoge en el acta número 4 del día 8 de marzo. En esta reunión acuden el tutor de la empresa y Yacy Lodos. En dicha reunión el tutor de la empresa valida lo realizado hasta el momento y se habla de las siguientes historias de usuario a seguir en el sprint 2.

Hasta ahora no hay desviaciones horarias, por lo que podemos decir que los 17 o 18 puntos de historia es una buena estimación para el tiempo de un sprint y no hay replanificación.

3.2. Sprint 2

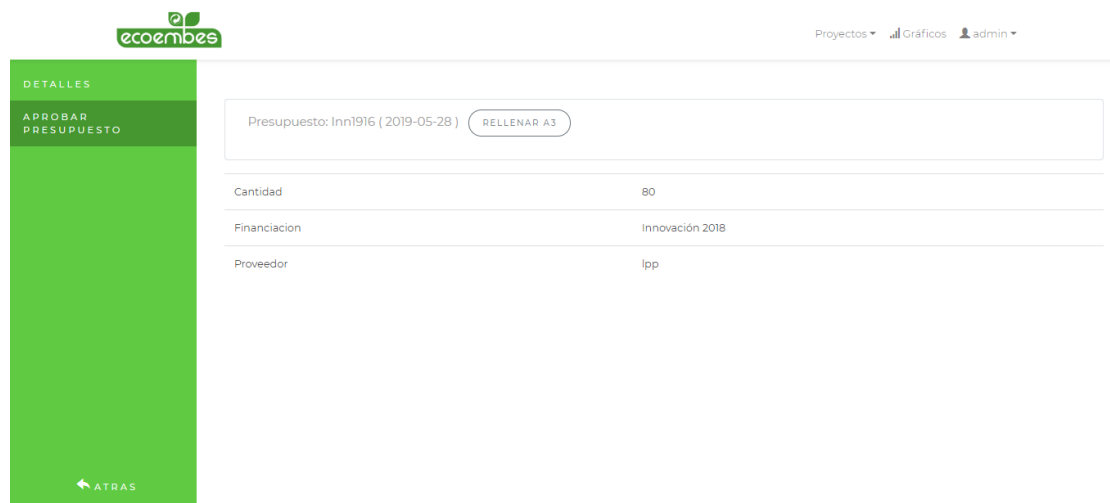
3.2.1. Incremento

Al final de este sprint deberíamos tener los siguientes incrementos:

1. Formulario para Aprobar presupuesto.
2. Botón para cambiar el estado del proyecto.
3. Funcionalidad de descarga del A3 en pdf.
4. Tablero Kanban.
5. Formulario para añadir/modificar tareas.

3.2.2. Trabajo de desarrollo

Para pasar un proyecto de estado “*en definición*” a estado “*en curso*”, tenemos que aprobar el presupuesto. Para poder aprobar un presupuesto diseñó una pantalla (ilustración 8) donde tendré detalles del presupuesto y un botón que al pulsar sobre él me lleve al formulario de aprobación/rellenar el A3.



Cantidad	80
Financiación	Innovación 2018
Proveedor	lpp

Ilustración 8: Aprobar proyecto

El formulario para rellenar el A3 (plantilla proporcionada por la empresa para aprobar un presupuesto), será un formulario que complete los datos del proyecto; es decir al añadir un nuevo proyecto no aparecen ciertos campos para rellenar, ya que muchos de ellos no quedarán bien definidos como: proveedores, equipo de proyecto, etc.

Por lo que diseñó este formulario con los datos de la tabla proyecto que hemos visto previamente y desarrolló un script para comprobar los datos, añadir los nuevos datos y cambiar el estado del proyecto a “*en curso*”. Tras el desarrollo previo realizó pruebas de funcionalidad. En este caso, compruebo que los datos se añaden correctamente a la base de datos, que se cambia el estado del proyecto en el tablero y que los datos que se listan son correctos.

Para cambiar el estado de los proyectos, añadió un desplegable (ilustración 9) con los posibles estados en las tarjetas de los proyectos, los proyectos en la columna *nuevas propuestas* tendrá la opción de pasar el proyecto a “*descartado*” y los proyectos en la columna de una *área temática* tendrá las opciones de “*en curso*”, “*stand by*”, “*finalizado*” o “*descartado*”; tras realizar

el diseño le proporcione funcionalidad, para ello desarrollo un script que modifique el estado del proyecto y lo liste en el tablero correctamente.

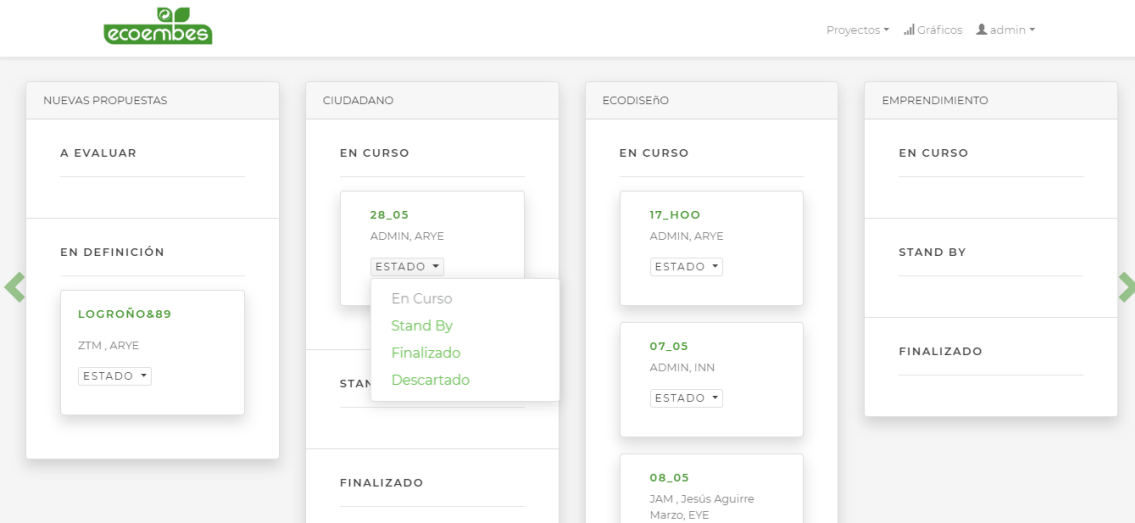


Ilustración 9: Tablero áreas con botón cambiar de estado.

Para realizar la descarga del A3 (ilustración 10) en pdf, busco una librería que me permita hacerlo de una manera sencilla, para ello escojo fpdf [9] es una clase escrita en php, gratuita y libre y ya la había utilizado con anterioridad. Tras estudiar cómo realizar la funcionalidad que necesito, la implemento; para ello genero un script que recoja todos los datos del A3 de un proyecto e introduzco los datos en la plantilla que me proporciona la empresa.

Esta clase fpdf es de gran ayuda, aunque tras usarla me doy cuenta de que es más costoso su uso de lo que pensaba, ya que la vez anterior no había tantos datos ni la estructura del documento era tan compleja. Ya que, para introducir los datos en la plantilla, hay que introducirlos por coordenadas.

Tras implementarlo realizo pruebas de que funciona correctamente y la descarga es la esperada.

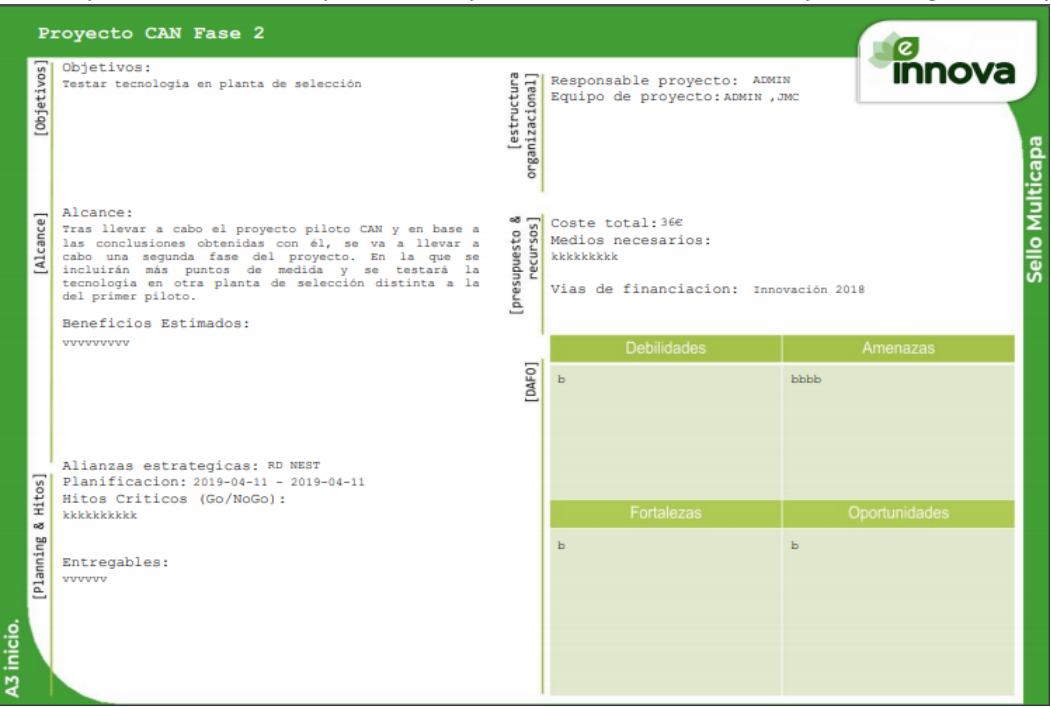


Ilustración 10: Documento A3.

Sigo con el diseño de la interfaz del Tablero Kanban, siguiendo las necesidades de la empresa y siguiendo el esquema Kanban general.

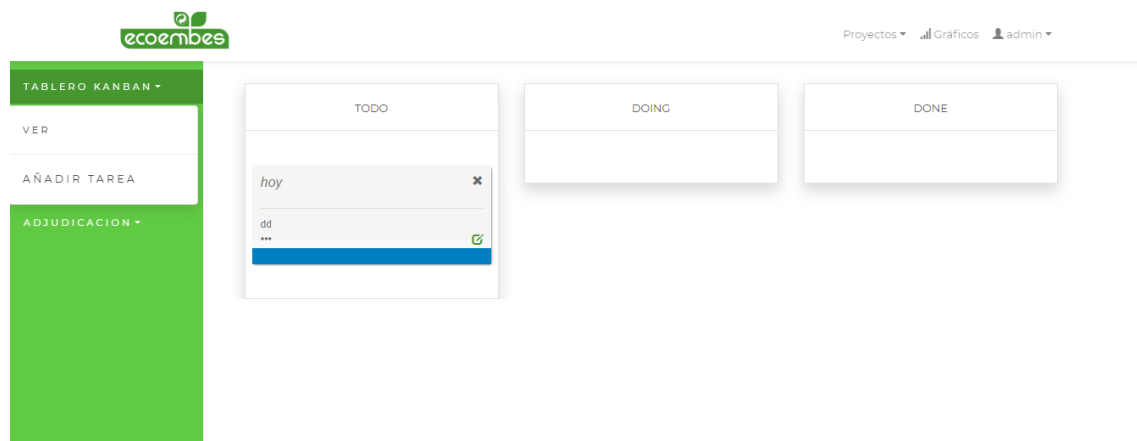


Ilustración 11: Tablero Kanban.

Tras el desarrollo del tablero, necesito poder añadir tareas, para ello diseño una tabla “Tarea”. Como vemos, la tabla “Tarea” tiene un *idTarea* que será incremental, *proyecto* el cual corresponde con el id del proyecto al cual pertenece la tarea, *prioridad* (alta, baja, normal), *estado* (do,doing,todo), *color* (el cual quieres que tenga la tarjeta).

nuevabd tarea	
idTarea	: int(45)
proyecto	: varchar(100)
empleado	: varchar(11)
nombre	: varchar(45)
detalle	: varchar(100)
prioridad	: varchar(11)
estado	: varchar(11)
color	: varchar(11)
fecha_ini	: date
fecha_fin	: date

Implemento la clase Tarea y añado en *GestorBD* los métodos necesarios para añadir una tarea, eliminar, modificar y listar. Después implemento un formulario para poder añadir la tarea. Este formulario recoge los datos necesarios para completar la base de datos y añado un script php que sirva para validar los datos del formulario y para añadir la tarea correctamente.

Tras esto realizo pruebas de funcionalidad para ver que en la base de datos se añade correctamente y que las tareas se listan correctamente en el tablero Kanban.

Para darle toda la funcionalidad al tablero implemento la funcionalidad de poder modificar una tarea, para poder cambiar las fechas o bien cambiar de estado la tarea. Para esto habilito un botón de edición, el cual habilitará el mismo formulario (*ilustración 12*) que hemos rellenado para crear la tarea, pero rellenado con los datos con los que añadimos la tarea previamente, con un botón para guardar. Tras guardar comprobaré los datos nuevos y modificaré los datos previos, así se podrán ver los datos actualizados en el tablero.

Ilustración 12: Formulario modificar tarea.

También debo permitir que las tareas se puedan eliminar. Para ello en las tarjetas de las tareas añado un botón cuya funcionalidad sea que la tarea se elimine inmediatamente para poder ver el tablero actualizado.

Finalmente realizo las pruebas de esto último, para ello compruebo que el formulario de edición está relleno con los datos correctos, modifico los datos y compruebo que se almacenan correctamente y que al pulsar sobre el botón de eliminar se elimina tanto del tablero como de la base de datos.

3.2.3. Sprint Review

Tras acabar este sprint, me reúno con mi tutor de la empresa para conocer el estado del proyecto y para planificar el siguiente sprint (*Anexo: Acta número 6, día 29 de marzo*). En esta reunión también se plantea realizar una reunión para comprobar los requisitos realizados (ver sprint Review del sprint 2) mediante una prueba funcional, es decir mediante el uso de la aplicación, el resultado es positivo, por lo que no hay desviaciones horarias.

3.3. Sprint 3

3.3.1. Incremento

Tras la realización de este sprint, deberíamos tener los siguientes incrementos:

1. Formulario para añadir/modificar una Adjudicación.
2. Funcionalidad de descarga de la Adjudicación.
3. Formulario para añadir/modificar nota interna.
4. Pantalla para subir documentos de los proveedores.
5. Lista de documentos de proveedores subidos.
6. Funcionalidad de descarga/eliminación de los documentos de los proveedores.
7. Añadir ampliación del presupuesto.

3.3.2. Trabajo de desarrollo

Comenzamos este sprint con la resolución de dudas sobre el proceso que siguen las adjudicaciones y facturaciones (*Anexo: acta número 7, día 2 de abril*). Una vez entendido el proceso (*ilustración 1, apartado 1.2.1 requisitos funcionales*) correctamente comienzo diseñando la tabla “Adjudicación”. Para ello la empresa me proporciona un documento (*ilustración 13*) el cual tengo que tomar como plantilla y usar para saber qué datos tengo que guardar y recoger en el formulario.

Periodo	Desde	Hasta
	27/12/2018	31/03/2019

Importe de la Solicitud de Pedido	Importe de la Orden de Pedido
18890,94 €	18890,94 €

¿Orden de Pedido presupuestada?	¿Urgencia o especificidad?
SI	Urgente

Justificación de Solicitud de Pedido NO Presupuestada o realizada por urgencia o especificidad
Lanzamiento de proyecto de Incentivos de forma urgente y proyecto específico.

Ofertante	Importe	¿Adjudicado?
Quickcargo SA	18890,94 €	SI
		Elige un elemento.

Ilustración 13: Documento adjudicación.

Para la creación de tablas y métodos, voy a seguir el proceso que he ido realizando en los anteriores sprints. Para ello creo la tabla “Adjudicación”, la clase *Adjudicación* y en la clase *GestorBD* añado los métodos necesarios.

nuevabd adjudicacion	
idAdjudicacion	: int(11)
idproy	: varchar(45)
ordenGasto	: varchar(20)
solicitudPedido	: varchar(20)
nOrdenPedido	: varchar(20)
ordenPedido	: varchar(45)
descripcion	: varchar(1000)
servicio	: varchar(45)
centroCoste	: varchar(45)
centroBeneficio	: varchar(100)
linea	: varchar(45)
material	: varchar(50)
periodo_ini	: date
periodo_fin	: date
importeSolicitudPedido	: varchar(50)
importeOrdenPedido	: varchar(50)
presupuestado	: varchar(4)
urgencia_especifico	: varchar(45)
justificacion	: varchar(45)
oferante	: varchar(45)
importe	: varchar(50)
adjudicado	: varchar(4)
descripcionOferante	: varchar(100)
proveedorHomologado	: varchar(4)
familia	: varchar(50)
comunicada	: varchar(4)
confidencialidad	: varchar(4)
ampliacion	: varchar(45)

Sigo con el diseño del formulario para permitir añadir la adjudicación. Para ello sigo con el estilo de formularios utilizado hasta el momento y le proporciono funcionalidad. Añado un nuevo script php, en el cual recojo todos los datos, genero el id de la adjudicación y añado la adjudicación.

Tras estos pasos, hago pruebas de funcionalidad, ya que al haber tantos datos puedo haberme equivocado al recoger algún dato del formulario o al añadirlos a la base de datos. Compruebo que en la base de datos esté todo correctamente añadido y que no se produce ningún error.

Tras ver que todo funciona correctamente, añado la funcionalidad de editar la adjudicación. Para ello habilito el formulario realizado en el paso anterior rellenado con los datos con los que hemos creado la adjudicación y al guardar los nuevos datos compruebo mediante un script los datos y los almaceno en la base de datos.

Para añadir la funcionalidad de descarga en word (.docx), busco una librería que me permita realizarlo de forma sencilla. Encuentro PHPWord [10], librería escrita en php que permite utilizar plantillas de documentos e insertar en estas plantillas variables (en el documento deseado

insertamos `${nombreVariable}`), las cuales mediante esta librería sustituiremos por nuestros datos. Escojo esta librería ya que no hay muchas en cuanto a descargas en Word y ya que es de las más utilizadas, por lo que tiene una buena documentación.

Tras estudiar cómo utilizar la librería en mi caso, desarrollo un script php que recoja los datos de la adjudicación y sustituya las variables del documento que me proporciona la empresa y ejecute la descarga del documento.

Pruedo el proceso de añadir una adjudicación nueva, la descargo y veo que esta se descarga correctamente. También compruebo que el proceso de editar la adjudicación previamente añadida y volver a descargarla, para ver si recoge los datos nuevos.

La empresa me proporciona la plantilla del documento Nota Interna (nota que resume el A3, *ilustración 13*). Primero diseño la tabla “*NotaInterna*”, la cual recoge los datos necesarios para rellenar el documento, implemento su clase y añado los métodos necesarios a la clase *GestorBD*.

nuevabd notainterna	
id	int(11)
idproy	varchar(20)
de	varchar(30)
para	varchar(30)
cc	varchar(30)
fecha	date
asunto	varchar(30)
ayj	varchar(1000)
objetivo	varchar(1000)
presupuesto	varchar(1000)
solicitud	varchar(1000)

Sigo con el diseño de formulario para añadir la nota interna y desarrollo un script para recoger los datos del formulario, verificarlos y añadir la nota interna. Tras añadirla nos mostrará el listado o las notas internas que hemos añadido y desde ahí podremos descargarla o editarla.

Para poder editar añadiré un botón, el cual habilitará el mismo formulario que hemos utilizado para añadir, pero con los datos guardados. Para la funcionalidad de descargar, vamos a usar la misma librería usada para la adjudicación (PHPWord) y básicamente seguiremos el mismo patrón explicado anteriormente con la plantilla de nota interna.

NOTA INTERNA

De: \$origen
 Para: \$destino
 CC: \$cc
 Fecha: \$fecha
 Asunto: \$asunto

Antecedentes y Justificación
 \$(antecedente)

Objetivo
 \$(objetivo)

Presupuesto y duración
 \$(presupuesto)

Solicitud de ofertas
 \$(solicitud)

Revisado por \$(0)	\$(2)
\$(origen)	\$(destino)

Ilustración 13: Documento nota interna.

Realizamos las pruebas de seguir el proceso de creación de una nota interna; la añadimos, vemos que se añade correctamente en la base de datos, la descargamos, vemos que los datos se muestran y son correctos y editamos y vemos que la base de datos se actualiza.

Seguimos con los documentos de los proveedores. La empresa me ha pedido que se puedan subir, eliminar y descargar (no editar debido a que estos documentos los rellena el proveedor). La empresa quiere poder acceder a todos los documentos de un proyecto desde una plataforma; en este caso mi aplicación.

Para ello vamos a implementar que cada vez que se cree un nuevo proyecto se cree una carpeta con su id en el servidor y que cada vez que se cree una adjudicación se cree una carpeta del id de la adjudicación en la carpeta del proyecto.

Para la funcionalidad realizamos un script que te permita la subida de un documento, y diseñamos una interfaz (*ilustración 14*). Diseñamos una pantalla en la cual tenemos un input para subir un documento y un listado de los documentos subidos; lo hacemos para cada adjudicación ya que los documentos de los proveedores son para cada adjudicación de un proyecto. En el listado de los documentos alojados en el servidor, vamos a tener un botón para descargar y otro para eliminar. Al pulsar sobre el botón eliminar se ejecutará un script php, que eliminará el fichero seleccionado y haremos lo mismo para el botón descargar.

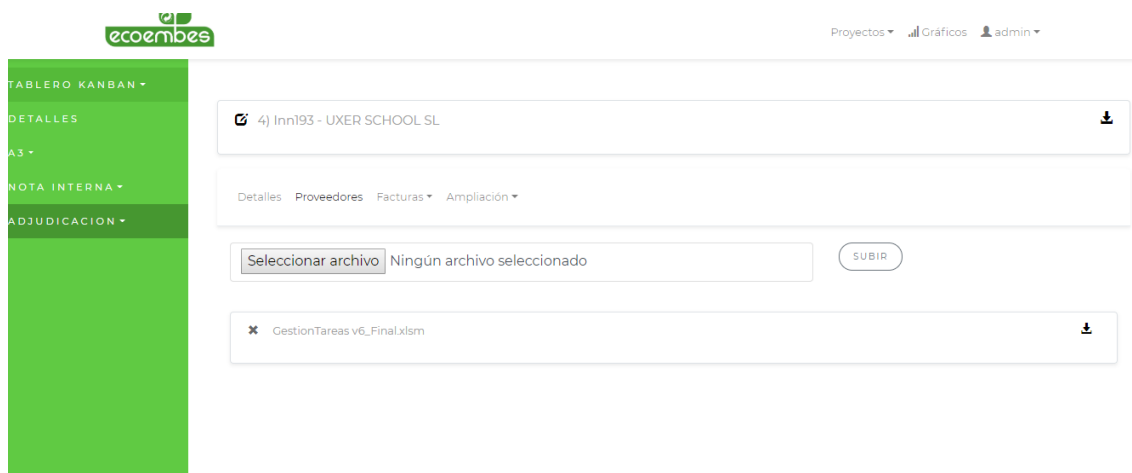


Ilustración 14: Subir archivos proveedores.

Realizo pruebas de subir un documento, veo que se lista correctamente y que está almacenado en el servidor donde debe, pruebo a descargarlo y a eliminarlo.

Para la ampliación de un presupuesto, el cliente pidió una copia de la adjudicación que queremos ampliar, pero con distinto presupuesto. En el caso que tengamos una adjudicación para un proveedor con “x” presupuesto, pero que durante el proyecto necesitamos más presupuesto, entonces se realiza una ampliación, la cual consiste en coger la adjudicación que queremos ampliar y hacer una copia, pero con otro importe, el importe que necesitamos de más.

Por lo que para realizar esta funcionalidad en la tabla “Adjudicación” añadimos la columna *ampliación*, en la cual voy a almacenar el id de la adjudicación que estoy ampliando; así podré listar dentro de una adjudicación qué ampliaciones hay. Básicamente lo que hago es habilitar el formulario que utilizo para añadir una adjudicación relleno con los datos de la adjudicación que queremos ampliar y la guardo como si fuera una nueva adjudicación, pero en la columna de ampliación almacenaré el *id* de la adjudicación que amplio.

Finalmente realizo pruebas de que funciona correctamente y no se produce ningún error.

3.3.3. Sprint Review

A mitad de este sprint se hizo una verificación de funcionalidad por parte del tutor de la empresa (*Anexo: Acta número 8, día 11 de abril*), en el cual se pidieron cambios en cuanto a algún término gramatical y de algún dato de la aplicación. Al final del sprint se verificó que todo acabara como se esperaba (*Anexo: Acta número 10, día 19 de abril*).

En este sprint puedo decir que no ha habido desviaciones horarias, aunque he añadido los cambios explicados anteriormente en el sprint, ya que no suponía mucho tiempo de dedicación.

3.4. Sprint 4

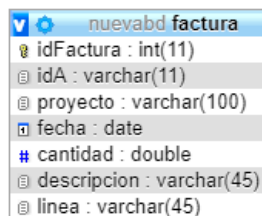
3.4.1. Incremento

Tras las tareas del cuarto sprint deberíamos tener los siguientes incrementos:

1. Formulario para añadir facturaciones.
2. Pantalla donde ver las facturaciones añadidas y donde eliminarlas.
3. Formulario para que un usuario añada su dedicación semanal.
4. Formularios tanto dentro como fuera de la aplicación para cambiar la contraseña.
5. Panel de gráficos con:
 - a. Gráfico de presupuesto por línea
 - b. Gráfico presupuesto de áreas temáticas de innovación.

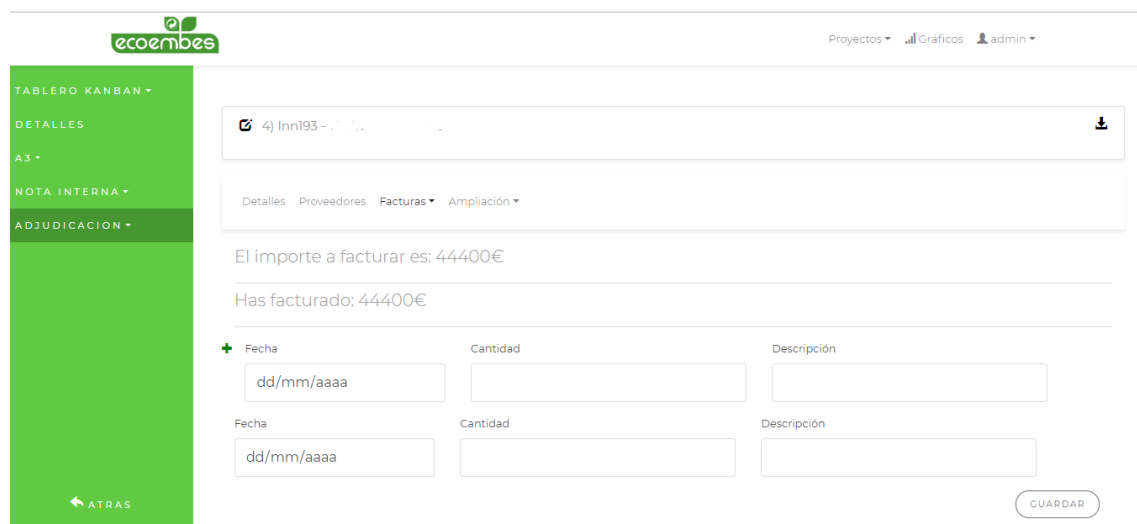
3.4.2. Trabajo de desarrollo

Comenzamos este sprint con el diseño de la tabla “Factura” ya pensando en el desarrollo y las necesidades que nos van a surgir a la hora del desarrollo de los gráficos. Siguiendo el proceso de los anteriores sprints implemento la clase Factura y los métodos necesarios en GestorBD.



nuevabd factura	
idFactura	: int(11)
idA	: varchar(11)
proyecto	: varchar(100)
fecha	: date
cantidad	: double
descripcion	: varchar(45)
linea	: varchar(45)

Tras tener la parte funcional, diseño la parte de interfaz (*ilustración 15*). Para ello realizo un formulario con inputs dinámicos. Estos inputs se generarán tras pulsar un botón, añadiendo así los campos necesarios del formulario, debido a que puedes tener varias fechas para facturar una adjudicación y no siempre van a ser iguales.



The screenshot shows a web application interface for adding invoices. At the top, there's a header with the 'ecoembes' logo and navigation links for 'Proyectos', 'Gráficos', and 'admin'. A sidebar on the left contains a 'TABLERO KANBAN' and a 'DETALLES' section with a sub-menu: 'A3', 'NOTA INTERNA', and 'ADJUDICACION'. The main content area is titled '4) Inn193 - ...' and includes a download icon. Below this, there are tabs for 'Detalles', 'Proveedores', 'Facturas' (selected), and 'Ampliación'. The 'Facturas' tab shows the 'El importe a facturar es: 44400€' and 'Has facturado: 44400€'. A green sidebar on the left has a '+ ATRAS' button. The main form area has a green header with a '+' icon and three columns: 'Fecha', 'Cantidad', and 'Descripción'. There are two rows of input fields for these columns. A 'GUARDAR' button is at the bottom right.

Ilustración 15: Añadir facturas.

Habiendo añadido la funcionalidad a la interfaz, compruebo que la funcionalidad de añadir funciona correctamente. Después incorporo en otra pantalla las facturaciones (*ilustración 16*) que hemos realizado con un botón para poder eliminarlas en caso de equivocación. Tras esto

compruebo que se elimina completamente de la base de datos y que se actualizan los datos de facturación.

El importe a facturar es: 44400€

Fecha	Cantidad	Descripción
28/02/2019	22200	
04/04/2019	6660	
01/07/2019	13320	
22/07/2019	2220	

Has facturado: 44400€

Ilustración 16: Ver y eliminar facturas.

La siguiente tarea consiste en añadir la dedicación de usuario. Para ello añadido a la base de datos la tabla “Dedicación”.

nuevabd dedicacion	
usr	: varchar(45)
semana	: varchar(10)
innovacion	: varchar(10)
incentivos	: varchar(10)
visitas	: varchar(10)
presentacion	: varchar(10)
estudiosOperativos	: varchar(10)
formacion	: varchar(10)

La cual contiene *usr*, al que pertenece dicha dedicación semanal. *semana*, la cual hemos incorporado esta dedicación. *usr* y *semana* deben ser únicos, un usuario en una semana solo puede introducir los datos una vez. Añadimos los demás datos relativos a los campos solicitados por el cliente necesarios en cuanto a la dedicación.

Seguimos con la implementación de la clase Dedicación e incorporamos al fichero *GestorBD* los métodos necesarios para añadir y para comprobar si un usuario ya ha introducido en una semana concreta su dedicación.

Implementamos la interfaz del formulario (*ilustración 17*) y el script que validará los datos y los almacenará en la base de datos. El formulario está compuesto de los campos que nos pidió el cliente en los requisitos funcionales para la dedicación de usuario. Esta dedicación de usuario es necesaria para ver en que ámbitos de trabajo invierten más tiempo cada empleado semanalmente.

Ilustración 17: Formulario dedicación de usuario.

En esta parte del usuario en la aplicación, incorporamos también la interfaz para poder modificar la contraseña. Como dijimos en el primer sprint, en la tabla empleado tenemos un campo palabraClave para comprobar que el usuario es quien dice ser. Este cambio de contraseña se va a poder hacer desde el Login (si se nos ha olvidado la contraseña) o desde dentro de la aplicación (si queremos cambiarla). Para esta funcionalidad añadido al *GestorBD* el método necesario para cambiar la contraseña y para la comprobación de que la palabra clave es de un usuario en concreto.

Desde dentro de la aplicación, no necesitaremos la palabraClave, ya que hemos accedido a la aplicación por nuestras claves, pero sí lo mostramos en caso de que quiera recordarlo.

Ilustración 18: Formulario cambio contraseña, dentro de la aplicación.

Desde fuera de la aplicación se hará en dos pasos (*ilustración 19 e ilustración 20*); uno para comprobar que el usuario y la palabra clave es correcta y el otro para añadir la nueva contraseña en caso de que la verificación anterior sea correcta.

Ilustración 19: Cambio contraseña, fuera de la aplicación.

Ilustración 20: Cambio contraseña, fuera de la aplicación.

Comprobamos que esta funcionalidad de cambio de contraseña funciona correctamente, tanto fuera de la aplicación, como, dentro y muestra los correspondientes mensajes de error en caso de haberlos.

Pasamos a implementar el panel de gráficos. El primer gráfico a realizar es el gráfico que compara de cada mes de un año, cuánto se ha adjudicado frente a cuanto se ha facturado con una línea continua representando el presupuesto anual de esta línea. Este gráfico lo tendremos que realizar para cada línea de proyectos (Innovación, Incentivos, Formación, Estudios operativos).

Para los gráficos utilizaré Google Charts [11], servicio web interactivo proporcionado por Google que crea gráficos a partir de datos que da el usuario. Para comenzar realizo una función en GestorBD que recoja los importes adjudicados de cada mes de una línea y otra para que recoja los datos de facturación y que nos devuelvan estos datos. Cada mes deberá ser incremental, es decir, que tenga lo adjudicado y facturado ese mes más los anteriores.

Para rellenar el gráfico, utilizo XMLHttpRequest para enviar los datos mediante peticiones POST y recibir datos de tipo Json. Esto lo hago ya que no nos interesa recargar la página cada vez que cambie la opción del select (*ilustración 21*).

```

function lineasMes(){
    var dato = "l="+document.getElementById("lineas").
    options[document.getElementById("lineas").selectedIndex].value;

    //Envio por post del dato seleccionado a verifica.php
    var xhr = new XMLHttpRequest();
    xhr.open("POST", "verificaLinea.php");
    xhr.setRequestHeader("Content-type","application/x-www-form-
    urlencoded");
    xhr.send(dato+"&action=linea");
    //devuelve un array que se incluye en el select b
    xhr.onreadystatechange = function(){
        if(xhr.readyState == 4 && xhr.status == 200){
            var result=JSON.parse(xhr.responseText);

            if(result!="KO"){
                google.charts.load('current', {'packages':['corechart']});
                google.charts.setOnLoadCallback(lineasM);

                function lineasM() {
                    var data = google.visualization.arrayToDataTable([
                        ['Mes', 'Adjudicacion', 'Facturacion'],
                        ['ene.', result[0][1],result[1][1]],
                        ['feb.', result[0][2],result[1][2]],
                        ['mar.', result[0][3],result[1][3]],
                        ['abr.', result[0][4],result[1][4]],
                        ['may.', result[0][5],result[1][5]],
                        ['jun.', result[0][6],result[1][6]],
                        ['jul.', result[0][7],result[1][7]],
                        ['ago.', result[0][8],result[1][8]],
                        ['sep.', result[0][9],result[1][9]],
                        ['oct.', result[0][10],result[1][10]],
                        ['nov.', result[0][11],result[1][11]],
                        ['dic.', result[0][12],result[1][12]]
                    ]);

                    var options = {
                        title : '',
                        vAxis: {
                            minValue:0,
                            format: '#\''€\'',
                            gridlines: {count: 8}
                        },
                        colors:['#299D3B','#083281'],
                        seriesType: 'bars',
                        gridlines: {count:10}
                    };

                    var chart = new
                    google.visualization.ComboChart(document.getElementById('l
                    in'));
                    chart.draw(data, options);}
                }else{
                    document.getElementById("lin").innerHTML="";
                }
            }
        }
    }
}

```

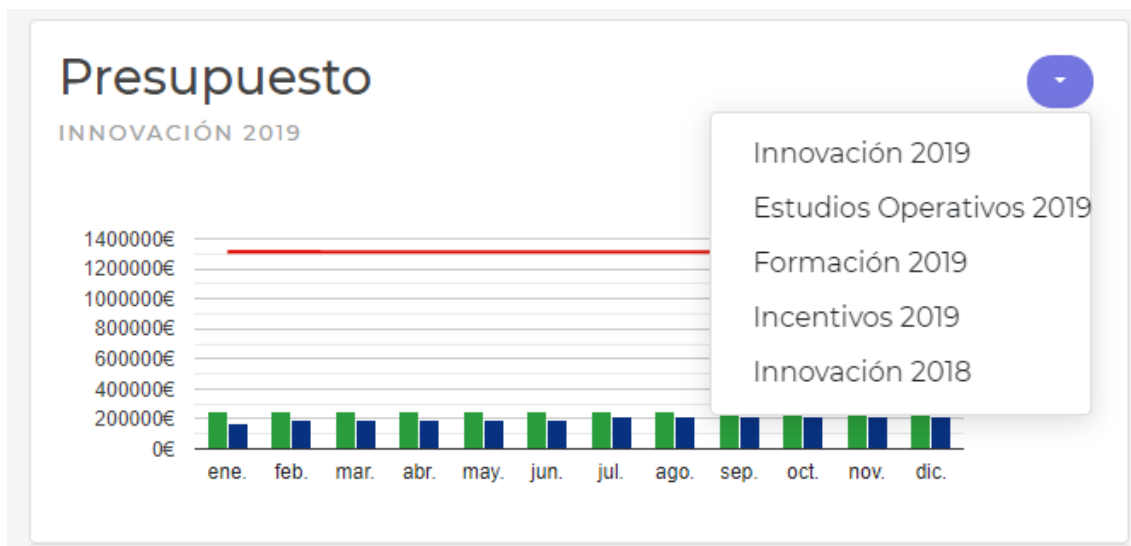


Ilustración 21: Gráfico presupuesto líneas,

Para el siguiente gráfico, que compara adjudicaciones y facturaciones por cada mes de las áreas temáticas de la línea innovación. Para el desarrollo sigo el mismo proceso, exceptuando que en este gráfico no va a haber la variable constante de presupuesto anual.



Ilustración 22: Gráfico presupuesto áreas innovación.

3.4.3. Sprint Review

Se realiza una reunión para validar lo realizado hasta el momento (*acta número 11, día 12 de mayo*). En esta reunión está presente el tutor de la empresa y Yacy Lodos. En dicha reunión el tutor de la empresa valida lo realizado hasta el momento, planificamos el último sprint y propone en el último sprint realizar Dailys meeting con el fin de ver los errores/cambios a realizar.

En cuanto a la variación de tiempo planificado y el tiempo real, en este sprint el último gráfico realizado costó menos tiempo del planificado, por lo que dio tiempo de subir la aplicación al

servidor de pruebas y a realizar pruebas de funcionalidad. El servidor lo proporciona el equipo de sistemas con todo instalado y preparado para las pruebas, con base de datos, apache y sftp.

3.5. Sprint 5

3.5.1. Incremento

Tras las tareas del quinto y último sprint deberíamos tener los siguientes incrementos:

1. Panel de gráficos con:
 - a. Gráfico porcentaje de dedicación por empleado.
 - b. Gráfico porcentaje dedicación total de los empleados.
 - c. Gráfico porcentaje estados proyecto.
 - d. Gráfico porcentaje adjudicado y facturado sobre un presupuesto.
2. Formulario para editar perfil.
3. Formulario para añadir proveedores.

3.5.2. Trabajo de desarrollo

En este sprint seguimos ampliando el panel de gráficos. El primero a realizar es el gráfico de dedicación (*ilustración 23*) para cada empleado. Para su desarrollo realizo una función que calcule el porcentaje en función de cada tema (innovación, incentivos, formación, estudios operativos, presentaciones y visitas) y en función de las horas semanales teniendo en cuenta que en verano son 35 y si no 40. Para introducir los datos en el gráfico lo realizo como en el anterior sprint.

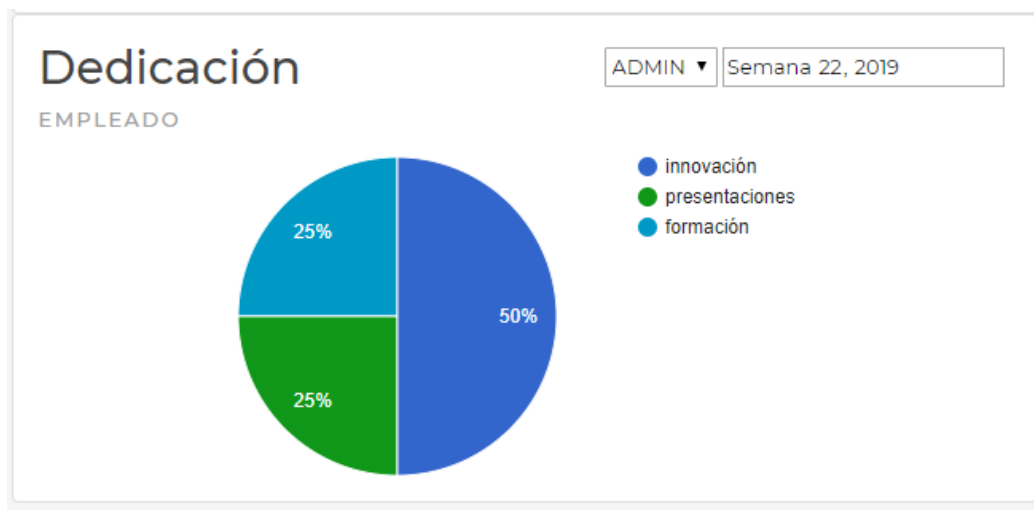


Ilustración 23: Gráfico porcentaje dedicación usuario.

Para el gráfico de estados de un proyecto vamos a seguir los pasos seguidos hasta ahora. Y para acabar con el panel de gráficos introducimos el último gráfico. El gráfico de porcentaje de un presupuesto (*ilustración 24*) con porcentajes de adjudicación y facturación frente al presupuesto. Para ello siguiendo con las pautas anteriores creamos una función que nos devuelva los porcentajes e introducimos estos datos en el gráfico.

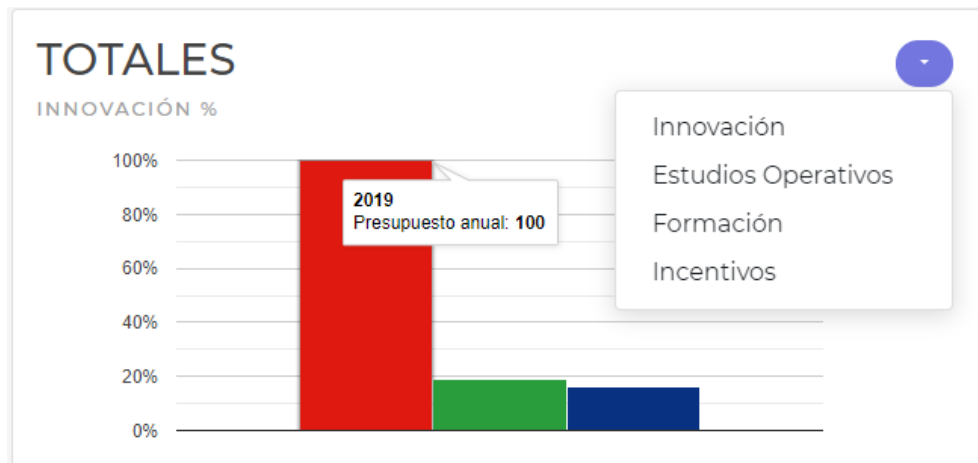


Ilustración 24: Gráfico porcentaje presupuesto proyecto.

Al finalizar el desarrollo de este gráfico, hago pruebas de funcionalidad; que los datos que recogen los gráficos son exactos, se cambian unos gráficos por otros al cambiar de opción, etc.

Continuamos con un formulario en la parte del usuario, que le permita modificar sus datos. Para ello implemento el método para modificar datos de un usuario en GestorBD, diseño la interfaz del formulario y le proporciono funcionalidad, recogiendo los datos, verificándolos y guardándolos o mostrando su respectivo mensaje de error.

Para añadir un proveedor diseñamos la tabla “Proveedores” y creamos su respectiva clase, Proveedor, con los datos que pide el tutor de la empresa. Entre ellos cabe destacar que tendremos como id el *nombre* correspondiente con el proveedor.

nuevabd	proveedores
personaContacto	: varchar(45)
nombre	: varchar(45)
email	: varchar(45)
tlfn	: int(11)
contacto_ecoembes	: varchar(45)

En *GestorBD* incorporamos el método de añadir proveedor y de listar los proveedores añadidos. Sigo con el diseño del formulario el cual refleja los datos de la tabla, este formulario va a ser una ventana emergente, para que se pueda añadir un

proveedor desde cualquier sitio.

Para la funcionalidad de guardado lo vamos a hacer mediante una petición Ajax, ya que al ser una ventana emergente nos interesa quedarnos en la página actual donde estamos, por lo que, recogemos los datos del formulario y al añadir se muestra una alerta de proveedor añadido o en caso de error una alerta refiriéndose al error producido.

En este sprint se había reservado tiempo para cambios, debido a que la aplicación se puso en uso por el cliente (para pruebas y verificación de cumplimiento de requisitos). Dando lugar a posibles cambios por parte del tutor de la empresa/ cliente. Entre ellos encontramos los siguientes:

- Cada vez que el presupuesto sea menor de 15.000 euros no habrá que realizar el A3 así que el proyecto pasará de estado “a evaluar” a estado “en curso”.
- Los detalles del proyecto deben ser editables, para dar mayor posibilidad de cambios.
- Solo se debe permitir hacer una adjudicación y en ella introducir todos los costes de proveedores. (Antes se debía hacer una por proveedor).
- Eliminar datos de prueba de la base de datos e introducir datos reales.

3.5.3. Sprint Review

Al final de este sprint se hace una reunión (*acta número 12, día 31 de mayo*) de validación y de presentación del proyecto. En ella se verifican que todos los requisitos planteados para este proyecto han sido cumplidos.

Aunque se proponen mejoras o una ampliación de la aplicación como pueden ser:

- La parte de administrador descartada al principio del proyecto por falta de tiempo (exclusiones).

4. Revisión de la gestión

Como hemos ido viendo en los sprint Review no ha habido casi ninguna desviación de tiempo. Por lo que la planificación y gestión de tiempo se pueden considerar como buenas. En cuanto a lo esperado de la aplicación y cumplimiento de requisitos, ha sido correcto (verificado por el cliente) y hemos alcanzado lo planificado.

5. Conclusiones

En este proyecto he desarrollado una aplicación de gestión de proyectos I+D funcional, la cual permite gestionar un proyecto desde su inicio hasta su fin. Permite gestionar presupuestos, gestionar tareas mediante un tablero Kanban, gestionar documentos internos, gestionar dedicación de empleados y ver un tablero con todas las estadísticas de forma visual (gráficos). Esta aplicación se va a llevar a producción para que los empleados de Ecoembes la utilicen en su día a día.

En este proyecto he aprendido la importancia de una buena planificación para la correcta realización de un proyecto a corto plazo, algo que aprendes en Taller Transversal II.

En cuanto a la parte de programación he aprendido el lenguaje PHP, Smarty y Bootstrap (para el diseño). La curva de aprendizaje de PHP ha podido ser más rápida gracias a la asignatura Programación de aplicaciones web. En cuanto a Smarty y Bootstrap gracias a los cursos realizados durante el proyecto.

Gracias a realizar el TFG en Ecoembes he podido conocer la metodología de trabajo de dicha empresa, la cual me ha aportado aptitudes de trabajo en equipo.

Tras la terminación del TFG, la empresa Ecoembes me contrató para completar la funcionalidad de la aplicación y su puesta en funcionamiento, lo cual considero que también es un respaldo al trabajo realizado y motivo de satisfacción personal.

Bibliografía

- [1] Ecoembes. Descripción del centro TheCircularLab.
<https://www.thecircularlab.com/descubre/que-es-thecircularlab/>
- [2] Trello. Herramienta analizada para el análisis de mercado.
<https://trello.com/>
- [3] Jira. Herramienta analizada para el análisis de mercado.
<https://es.atlassian.com/software/jira>
- [4] Wrike. Herramienta analizada para el análisis de mercado.
<https://www.wrike.com/es/tour/>
- [5] The Scrum Guide. Explicación metodología ágil y Scrum.
<https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf>
- [6] Bootstrap. Diseño interfaz de usuario.
<https://getbootstrap.com/docs/3.3/getting-started/>
- [7] w3schools. Curso sobre Bootstrap.
<https://www.w3schools.com/bootstrap4/>
- [8] Smarty. Uso en el desarrollo del front end de la aplicación.
<https://www.smarty.net/>
- [9] Librería FPDF. Descarga del documento de A3.
<http://www.fpdf.org/>
- [10] Librería PHPWord. Descarga de documentos de Adjudicación y nota interna.
<https://phpword.readthedocs.io/en/latest/>
- [11] Google charts. Desarrollo de gráficos de la aplicación.
<https://developers.google.com/chart/>

Anexo

El anexo va a estar compuesto de las actas de reuniones realizadas durante el trabajo fin de grado. Comenzando en la página siguiente.

	<h1>Acta de reunión</h1>	
---	--------------------------	---

Número:	1
Proyecto:	Aplicación web para la gestión de proyectos de I+D
Fecha:	04 de febrero de 2019
Lugar:	The circular Lab
Hora inicio:	12:00
Hora finalización:	14:00
Asistentes:	Yacy Lodos Saranova Jose Luis Moreno de Castro(Ecoembes) Cristina Serrano (Ecoembes)
Orden del día:	Recopilación de los requisitos iniciales de la aplicación y la metodología a seguir durante el proyecto.

Temas tratados:

1. Como es el proceso que siguen en la gestión de proyectos, para poder recoger los requisitos del cliente.
2. Entorno de trabajo.
3. Sistema de control de versiones.
4. Metodología de trabajo.

Temas pendientes

A continuación se listan los temas que quedan pendientes de realizar, los responsables de hacerlo y la fecha prevista para ello.

Tema	Responsables	Fecha resolución
Realizar curso sobre Scrum	Yacy Lodos Saranova	Semana del 04/02/19
Recoger los requisitos en un documento.		

	<h1>Acta de reunión</h1>	
---	--------------------------	---

Número:	2
Proyecto:	Aplicación web para la gestión de proyectos de I+D
Fecha:	06 de febrero de 2019
Lugar:	The circular Lab
Hora inicio:	12:00
Hora finalización:	13:00
Asistentes:	Yacy Lodos Saranova Jose Luis Moreno de Castro(Ecoembes) Cristina Serrano (Ecoembes)
Orden del día:	Verificación de los requisitos de la aplicación recogidos en la reunión anterior.

Temas tratados:

1. Verificación de los requisitos.
2. Resolución de dudas.

Temas pendientes

A continuación se listan los temas que quedan pendientes de realizar, los responsables de hacerlo y la fecha prevista para ello.

Tema	Responsables	Fecha resolución
Realizar las historias de usuario.	Yacy Lodos Saranova	Semana del 06/02/19
Revisar los requisitos.		

	<h1>Acta de reunión</h1>	
---	--------------------------	---

Número:	3
Proyecto:	Aplicación web para la gestión de proyectos de I+D
Fecha:	5 de marzo de 2019
Lugar:	Edificio CCT
Hora inicio:	14:00
Hora finalización:	15:00
Asistentes:	Yacy Lodos Saranova Jesús María Aransay Azofra.
Orden del día:	Revisión de la memoria y resolución de dudas sobre la documentación.

Temas tratados:

1. Revisión de la memoria.
2. Resolución de dudas.

Temas pendientes

A continuación se listan los temas que quedan pendientes de realizar, los responsables de hacerlo y la fecha prevista para ello.

Tema	Responsables	Fecha resolución
Añadir requisitos funcionales.	Yacy Lodos Saranova	Semana del 06/03/19
Revisar y modificar pila de producto.		
Revisar documentación de la metodología de trabajo.		

	<h1>Acta de reunión</h1>	
---	--------------------------	---

Número:	4
Proyecto:	Aplicación web para la gestión de proyectos de I+D
Fecha:	8 de marzo de 2019
Lugar:	The circular Lab
Hora inicio:	12:00
Hora finalización:	14:00
Asistentes:	Yacy Lodos Saranova Cristina Serrano (Ecoembes)
Orden del día:	Revisión de lo realizado en el primer Sprint.

Temas tratados:

1. Revisión del Sprint 1.
2. Historias de usuario para el sprint 2.

Temas pendientes

A continuación se listan los temas que quedan pendientes de realizar, los responsables de hacerlo y la fecha prevista para ello.

Tema	Responsables	Fecha resolución
actualizar pila de producto	Yacy Lodos Saranova	Semana del 11/03/19

	<h1>Acta de reunión</h1>	
---	--------------------------	---

Número:	5
Proyecto:	Aplicación web para la gestión de proyectos de I+D
Fecha:	18 de marzo de 2019
Lugar:	Edificio CCT.
Hora inicio:	13:00
Hora finalización:	14:00
Asistentes:	Yacy Lodos Saranova Jesús María Aransay Azofra.
Orden del día:	Revisión de la documentación.

Temas tratados:

1. Revisión de la documentación, posibles mejoras.
2. Estado del proyecto.

Temas pendientes

A continuación se listan los temas que quedan pendientes de realizar, los responsables de hacerlo y la fecha prevista para ello.

Tema	Responsables	Fecha resolución
Corregir errores en la memoria	Yacy Lodos Saranova	Semana del 22/03/19
Añadir en la memoria un EDT y un Gantt		

	<h1>Acta de reunión</h1>	
---	--------------------------	---

Número:	6
Proyecto:	Aplicación web para la gestión de proyectos de I+D
Fecha:	29 de marzo de 2019
Lugar:	The circular Lab
Hora inicio:	12:00
Hora finalización:	14:00
Asistentes:	Yacy Lodos Saranova Cristina Serrano (Ecoembes)
Orden del día:	Planteamiento del sprint 3 y conocer el estado del proyecto.

Temas tratados:

1. Ver que historias de usuario se van a realizar en el tercer sprint.
2. Conocer el estado del proyecto.
3. Planificar una reunión para verificar la aplicación realizada hasta el momento.

Temas pendientes

A continuación se listan los temas que quedan pendientes de realizar, los responsables de hacerlo y la fecha prevista para ello.

Tema	Responsables	Fecha resolución
Planificar el sprint 3	Yacy Lodos Saranova	Semana del 01/04/19
Documentar el sprint 2		

	<h1>Acta de reunión</h1>	
---	--------------------------	---

Número:	7
Proyecto:	Aplicación web para la gestión de proyectos de I+D
Fecha:	02 de abril de 2019
Lugar:	The circular Lab
Hora inicio:	12:00
Hora finalización:	14:00
Asistentes:	Yacy Lodos Saranova Cristina Serrano (Ecoembes)
Orden del día:	Resolución de dudas sobre el procedimiento de adjudicaciones y facturaciones.

Temas tratados:

1. Resolución de dudas sobre el proceso de adjudicaciones y facturaciones.
2. Conocer en qué estado está el sprint .

Temas pendientes

A continuación se listan los temas que quedan pendientes de realizar, los responsables de hacerlo y la fecha prevista para ello.

Tema	Responsables	Fecha resolución
Recoger el proceso en la aplicación de forma correcta.	Yacy Lodos Saranova	Semana del 02/04/19
Documentar el proceso.		

	<h1>Acta de reunión</h1>	
---	--------------------------	---

Número:	8
Proyecto:	Aplicación web para la gestión de proyectos de I+D
Fecha:	11 de abril de 2019
Lugar:	The circular Lab
Hora inicio:	10:00
Hora finalización:	13:30
Asistentes:	Yacy Lodos Saranova Cristina Serrano (Ecoembes)
Orden del día:	Revisión de los incrementos realizados hasta el momento.

Temas tratados:

1. Utilización de la aplicación.
2. Verificación de los requisitos realizados.
3. Cambios a realizar en la aplicación.
4. Planificación de los siguientes Sprints.

Temas pendientes

A continuación se listan los temas que quedan pendientes de realizar, los responsables de hacerlo y la fecha prevista para ello.

Tema	Responsables	Fecha resolución
Eliminación de datos que ya no utilizan	Yacy Lodos Saranova	Semana del 15/04/19
Añadir datos reales en la aplicación. (Empleados,etc)		
Realizar documentación de cambios.		

	<h1>Acta de reunión</h1>	
--	--------------------------	--

Número:	9
Proyecto:	Aplicación web para la gestión de proyectos de I+D
Fecha:	12 de abril de 2019
Lugar:	Edificio CCT
Hora inicio:	14:00
Hora finalización:	15:00
Asistentes:	Yacy Lodos Saranova Jesús María Aransay Azofra.
Orden del día:	Revisión de la documentación.

Temas tratados:

1. Revisión del documento.
2. Estado del proyecto.

Temas pendientes

A continuación se listan los temas que quedan pendientes de realizar, los responsables de hacerlo y la fecha prevista para ello.

Tema	Responsables	Fecha resolución
Corregir errores en la memoria	Yacy Lodos Saranova	Semana del 15/04/19
Añadir en la memoria apartado Entregables.		

	<h1>Acta de reunión</h1>	
---	--------------------------	---

Número:	10
Proyecto:	Aplicación web para la gestión de proyectos de I+D
Fecha:	19 de abril de 2019
Lugar:	The circular Lab
Hora inicio:	10:00
Hora finalización:	13:30
Asistentes:	Yacy Lodos Saranova Cristina Serrano (Ecoembes)
Orden del día:	Revisión del tercer sprint

Temas tratados:

1. Ver que el sprint ha acabado como se esperaba.
2. Ver historias de usuario que vamos a realizar en el cuarto y las posibles del quinto sprint.

Temas pendientes

A continuación se listan los temas que quedan pendientes de realizar, los responsables de hacerlo y la fecha prevista para ello.

Tema	Responsables	Fecha resolución
Planificación sprint 4	Yacy Lodos Saranova	Semana del 22/04/19
Documentación		

	<h1>Acta de reunión</h1>	
---	--------------------------	---

Número:	11
Proyecto:	Aplicación web para la gestión de proyectos de I+D
Fecha:	12 de mayo de 2019
Lugar:	The circular Lab
Hora inicio:	10:00
Hora finalización:	13:30
Asistentes:	Yacy Lodos Saranova Cristina Serrano (Ecoembes)
Orden del día:	Revisión del cuarto sprint

Temas tratados:

1. Ver que el sprint ha acabado como se esperaba.
2. Ver historias de usuario que vamos a realizar en el quinto sprint.

Temas pendientes

A continuación se listan los temas que quedan pendientes de realizar, los responsables de hacerlo y la fecha prevista para ello.

Tema	Responsables	Fecha resolución
Planificación sprint 5	Yacy Lodos Saranova	Semana de 13/05/19
Documentación		

	<h1>Acta de reunión</h1>	
---	--------------------------	---

Número:	12
Proyecto:	Aplicación web para la gestión de proyectos de I+D
Fecha:	31 de mayo de 2019
Lugar:	The circular Lab
Hora inicio:	10:00
Hora finalización:	13:30
Asistentes:	Yacy Lodos Saranova Cristina Serrano (Ecoembes) Jose Luis Moreno (Ecoembes)
Orden del día:	Revisión del quinto sprint

Temas tratados:

1. Presentación del proyecto.
2. Finalización del proyecto.